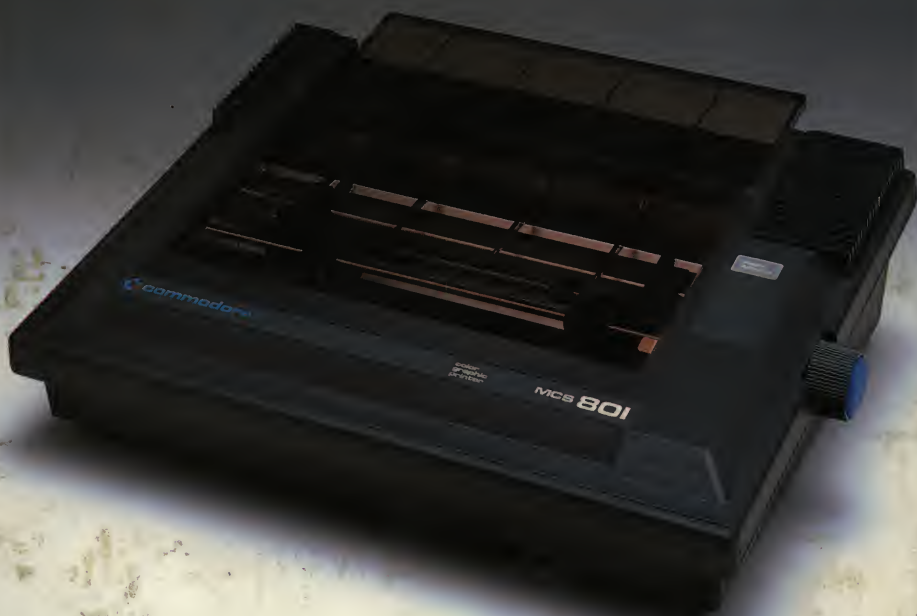


COMMODORE MCS-801 COLOR PRINTER

A Friendly Introduction to Your MCS-801 Color Printer



USER'S GUIDE

 **commodore**
COMPUTER

INFORMATION TO USER

"WARNING: THIS EQUIPMENT HAS BEEN CERTIFIED TO COMPLY WITH THE LIMITS FOR A CLASS B COMPUTING DEVICE, PURSUANT TO SUBPART J OF PART 15 OF FCC RULES. ONLY PERIPHERALS (COMPUTER INPUT/OUTPUT DEVICES, TERMINALS, PRINTERS, ETC.) CERTIFIED TO COMPLY WITH THE CLASS B LIMITS MAY BE ATTACHED TO THIS COMPUTER. OPERATION WITH NON-CERTIFIED PERIPHERALS IS LIKELY TO RESULT IN INTERFERENCE TO RADIO AND TV RECEPTION."

"THIS EQUIPMENT GENERATES AND USES RADIO FREQUENCY ENERGY AND IF NOT INSTALLED PROPERLY, THAT IS, IN STRICT ACCORDANCE WITH THE MANUFACTURER'S INSTRUCTIONS, MAY CAUSE INTERFERENCE TO RADIO AND TELEVISION RECEPTION. IT HAS BEEN TYPE TESTED AND FOUND TO COMPLY WITH THE LIMITS FOR A CLASS B COMPUTING DEVICE IN ACCORDANCE WITH THE SPECIFICATIONS IN SUBPART J OF PART 15 OF FCC RULES, WHICH ARE DESIGNED TO PROVIDE REASONABLE PROTECTION AGAINST SUCH INTERFERENCE IN A RESIDENTIAL INSTALLATION. HOWEVER, THERE IS NO GUARANTEE THAT INTERFERENCE WILL NOT OCCUR IN A PARTICULAR INSTALLATION. IF THIS EQUIPMENT DOES CAUSE INTERFERENCE TO RADIO OR TELEVISION RECEPTION, WHICH CAN BE DETERMINED BY TURNING THE EQUIPMENT OFF AND ON, THE USER IS ENCOURAGED TO TRY TO CORRECT THE INTERFERENCE BY ONE OR MORE OF THE FOLLOWING MEASURES:


- REORIENT THE RECEIVING ANTENNA
- RELOCATE THE COMPUTER WITH RESPECT TO THE RECEIVER
- MOVE THE COMPUTER AWAY FROM THE RECEIVER
- PLUG THE COMPUTER INTO A DIFFERENT OUTLET SO THAT COMPUTER AND RECEIVER ARE ON DIFFERENT BRANCH CIRCUITS

"IF NECESSARY, THE USER SHOULD CONSULT THE DEALER OR AN EXPERIENCED RADIO/TELEVISION TECHNICIAN FOR ADDITIONAL SUGGESTIONS. THE USER MAY FIND THE FOLLOWING BOOKLET PREPARED BY THE FEDERAL COMMUNICATIONS COMMISSION HELPFUL: 'HOW TO IDENTIFY AND RESOLVE RADIO-TV INTERFERENCE PROBLEMS.' THIS BOOKLET IS AVAILABLE FROM THE U.S. GOVERNMENT PRINTING OFFICE, WASHINGTON, D.C. 20402, STOCK NO. 004-000-00345-4."

PART NO: 320970

COMMODORE MCS-801 COLOR PRINTER USER'S GUIDE

A Friendly Introduction to Your MCS-801 Color Printer

 **commodore**
COMPUTER

COMMITTEE
FDG-50M
ARTISTS POLIO
JANUARY 1935

and the committee on the subject of the

of the committee
JANUARY

Table Of Contents

	Page No.
INTRODUCTION.....	1
CONGRATULATIONS.....	1
PRINTER FEATURES.....	1
CUSTOMER SUPPORT.....	2
CHAPTER 1: PREPARING TO USE YOUR PRINTER.....	3
A. UNPACKING YOUR PRINTER.....	3
B. MCS-801 PHYSICAL CHARACTERISTICS.....	4
C. INSTALLING THE PLASTIC RACK.....	6
D. LOADING THE PAPER.....	7
1. Continuous forms.....	7
2. Single sheets of paper.....	9
E. INSTALL AND REMOVE THE INK RIBBON CARTRIDGE.....	10
1. Installation.....	10
2. Removal.....	11
3. Inker Replacement.....	11
F. HEAD POSITION ADJUSTMENTS.....	13
G. CONNECTING THE PRINTER TO YOUR COMPUTER.....	14
H. PERFORMING THE POWER-ON TEST.....	16
I. PRINT TEST.....	16
J. TROUBLE SHOOTING.....	17
K. PRECAUTIONS.....	18
CHAPTER 2: USING YOUR PRINTER.....	19
A. INTRODUCTION.....	19
B. BASIC COMMANDS ASSOCIATED WITH YOUR PRINTER.....	20
1. The OPEN Command.....	20
2. The PRINT# Command.....	20
3. The CLOSE Command.....	20
4. The CMD Command.....	21
5. Command Example.....	21
B. PRINTING IN THE DIRECT MODE.....	23
1. Program List.....	24
C. PRINTING UNDER PROGRAM CONTROL.....	26
D. HOW TO PRINT, HOW TO SPECIFY COLOR.....	27
1. Text Printing and color specification.....	27
2. Special Character Printing.....	31
3. Scan Mode Bit Image Printing.....	40
E. ADVANCED PROGRAMMING TECHNIQUES THROUGH SECONDARY ADDRESSING.....	43
1. SA=0: Printing in Graphic Mode.....	44
2. SA=3: Setting the Number of Lines Per Page.....	47
3. SA=5: Defining a Programmable Character.....	51
4. SA=6: Setting Spacing Between Lines	55
5. SA=7: Selecting Business Mode.....	58
6. SA=10: Resetting the Printer.....	60
7. SA=19: Enable 2-pass Mode In Scan Mode Printing....	62
8. SA=20: Disable 2-pass Mode In Scan Mode Printing...	63
F. SPECIAL CONTROL CHARACTER FUNCTIONS.....	64
1. Enhance ON/OFF CHR\$(14)/CHR\$(15).....	65
2. Paging ON/OFF CHR\$(147)/CHR\$(19).....	66

3. Selecting Character Mode in local	
CHR\$(17)/CHR\$(145)	67
4. Reverse ON/OFF	CHR\$(18)/CHR\$(146) 69
5. Line Feed/Carriage Return/Carriage Stand	
CHR\$(10)/CHR\$(13)/CHR\$(141)	71
Automatic Carriage Return.....	72
6. Quotation Marks " or CHR\$(34)	74
7. Print Position Determination	
CHR\$(16);CHR\$(NH);CHR\$(NL)	77
8. Form Feed	CHR\$(12) 79
9. Repeat Graphic Printing.....	81
10. Dot Address Determination.....	82
APPENDICES.....	85
APPENDIX A: MCS-801 COLOR PRINTER SPECIFICATIONS.....	85
1. Printing Specifications.....	85
2. Paper feed Specifications.....	85
3. Inker ribbon Specifications.....	85
4. Operating Environments.....	85
APPENDIX B: CBM ASCII CODE TABLE & FONT TABLE.....	87
1. Graphic Mode.....	87
2. Business Mode.....	88
APPENDIX C: SERIAL INTERFACE INFORMATION.....	89
1. The Connector:	89
2. The Interface:	89
APPENDIX D: Hardcopy of the Text Screen.....	90
APPENDIX E: MCS-801 GENERAL TEST PROGRAM.....	92

INTRODUCTION

CONGRATULATIONS

We are certain that you will enjoy the convention and printer quality obtained from your purchase of this COMMODORE MCS-801. This User's Guide includes all the relevant information, necessary for your aid in it's use.

- Learn the names and functions of all external parts
- Install your Color Printer
- Connect the Color Printer to your computer
- Operate and maintain your Color Printer
- Write Programs for your Color Printer

The more familiar you become with COMMODORE BASIC SYNTAX the easier the programming of the printer, therefore it is advised to refer to either your computer's USER'S GUIDE and PROGRAMMER'S REFERENCE MANUAL.

Printer Features

This MCS-801 color dot matrix printer is designed to operate through software control. It prints colored characters, upper- and lower-case, alphabetic, numeric characters and all the graphic characters available on your COMMODORE computer, even a custom user-defined character. In addition, your color printer has bit image printing which enables hardcopy of a high-resolution screen.

Your Color Printer has the following features:

- Printing in eight different colors (including black and paper white)
- Standard COMMODORE serial interface to allow easy connection to other COMMODORE computer peripheral devices
- 80 column printing
- Prints 38 characters per second
- A form feed function
- The initial print position can be specified in either characters or dots.
- Both TRACTOR/FRICTION feed
- Colors can be specified in following two ways.
 1. Character units or 8-dot vertical graphic's columns.
 2. Single dot units scanning horizontally corresponding to the hammers in the print head.

Your Color Printer is designed to connect directly into your computer through the Serial Port (6 pin connector). It can be used with up to 4 Model 1541 floppy disk drives. These units are connected to the computer by "daisy chaining" the devices together -- each successive unit is plugged into the last unit.

Customer Support

Please check with your local COMMODORE dealer, COMMODORE User's Clubs and COMMODORE Magazines and POWER/PLAY for additional support for your new Color Printer. They can also keep you informed about possible application programs written for it.

CHAPTER 1
PREPARING TO USE YOUR PRINTER

A. UNPACKING YOUR PRINTER

Before you unpack your color printer, inspect the shipping carton for signs of damage. If it appears to be damaged, be especially careful when you inspect its contents. DON'T throw away any of the packaging material until you have located all the contents of the carton! The package should contain (Refer Figure 1).

1. Commodore Color Printer, Model MCS-801
2. Printer Ribbon Cartridge
3. Serial Cable
4. User's Guide
5. Warranty Card

If any of these is missing or damaged, notify your Commodore dealer immediately.

B. MCS-801 PHYSICAL CHARACTERISTICS

Front and Top View

Please note the following locations for MCS-801 features:

- * ON-OFF switch is on the left side of the printer housing.
- * A paper advance push switch incorporating a power indicator is located on the front right side of the printer housing.

<u>PARTS</u>	<u>OPERATION & FUNCTION</u>
a. Power Switch:	Turns printer ON and OFF
b. Power Indicator/ Error Indicator	Lights up when printer is turned on. Flash when an error is occurred. If the error is paper empty error, you can recover the error by pressing the PAPER ADVANCE PUSH SWITCH after paper reinsertion.
c. Printer Cover	Set your cover to protect your printer from dust and to reduce the noise level while the printer is in operation.
d. Paper Advance Knob	Once you have properly threaded your paper, you can use the Paper Advance Knob to move the paper (both forward and backward directions)
e. Paper Advance Switch	Use this switch to move the paper ahead, 1 line at a time in the forward direction only!

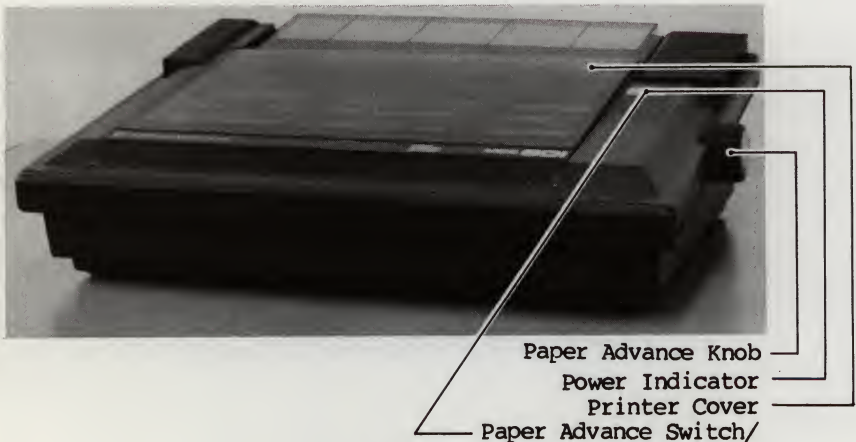


Figure 1.

Back View

Turn your Printer around so that back of the unit is facing you. On the left side you will see the power line connection situated just to the left of the fuse holder. On the right, you will see two serial interface jack socket, these are used to connect the unit to your computer.

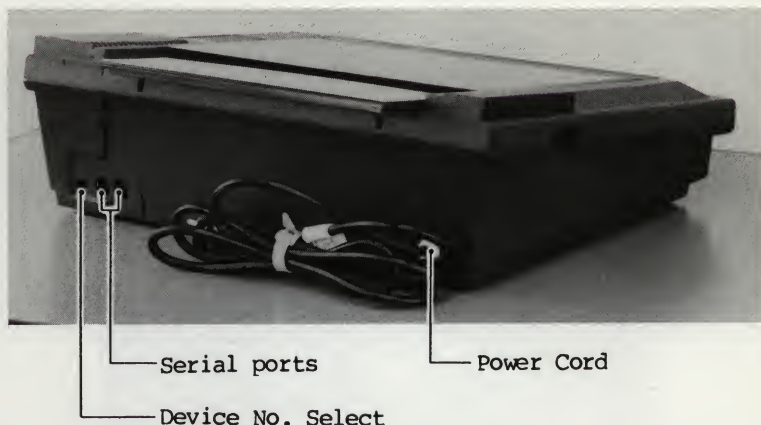


Figure 2.

- a. Power Line Connector Use the power cord supplied with your Color Printer.
- b. Serial Interfaces These jack sockets are used to connect your Color Printer to the computer. Use the Serial Interface cable supplied with the Color Printer. Refer to Chapter 1 section G for complete connection instructions.
- c. Self-Diagnostic Switch Select printer device number, "4" is the normal position for one printer. "5" and "6" are the position to use with your second or third printer.

C. INSTALLING THE PLASTIC RACK

Install the plastic rack by placing the ends of it into the holes situated in the housing (see Figure 3.).



Figure 3. Installation of the plastic rack

D. LOADING THE PAPER

1. Continuous forms



Figure 4.

- (1) Remove the printer cover by raising it towards you by about 90 degrees, and pull it off its pivot rest (Figure 4)



Figure 5.

- (2) Raise the plastic rack to a vertical position with respect to the case top, and remove it pivot rest. Move the head adjustment lever towards you. (Refer to [F] HEAD POSITION ADJUSTMENT)

- (3) Next lift up the friction roller bar and open tractor covers.



Figure 6.

- (4) Insert the paper from the rear, turn the paper advance knob so that the paper feeds around the platen and comes up in between the platen and print head.

- (5) Align the paper so that it will feed through



Figure 7.

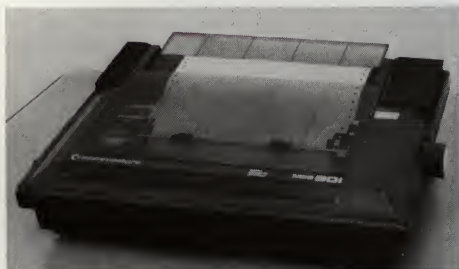


Figure 8.

consistently. Match the tractor teeth with the holes in the paper, close the tractor covers slide the right tractor slightly to the right in order to pull the paper taut. Lower the friction roller bar and establish that the two friction rollers are resting directly on top of the two large rubber rings.

- (6) Return the head adjustment lever to its original position. Replace the paper rack and printer cover in that order. Adjust the print position by turning paper advance knob.

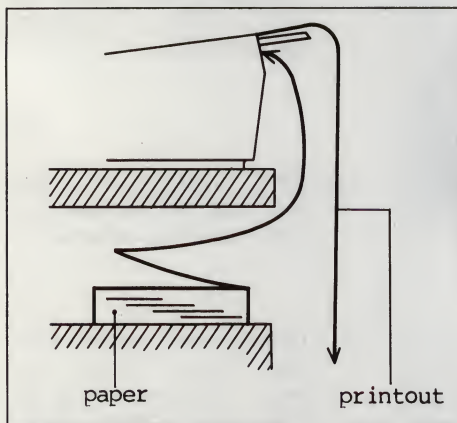


Figure 9.

(NOTE)

When using continuous forms, the paper supply should rest on the same table as the printer and the printed results should go to a lower position, such as the floor.

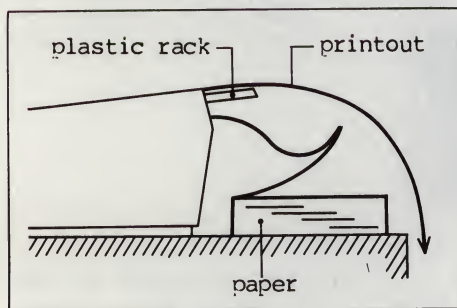


Figure 10.

2. Single sheets of paper

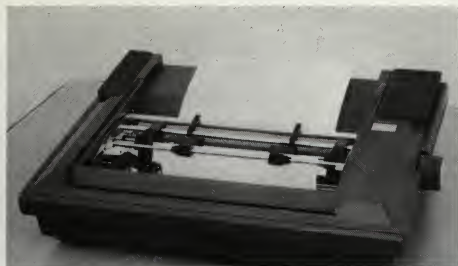


Figure 11.

For single sheets of paper follow the operations involving the tractors but with two conditions:

Note: 1. Because it is friction feed, be sure that the friction rollers are positioned directly on top of the rubber friction rollers and that these two sets of rollers properly grasp the paper.

2. A paper empty switch hidden behind the platen, when it does not sense paper the printer is switched off, therefore in order to print on the bottom two inches of a single sheet feed in about three inches of another sheet to cover the switch, this will compensate for the loss of paper.

E. INSTALL AND REMOVE THE INK RIBBON CARTRIDGE

1. Installation

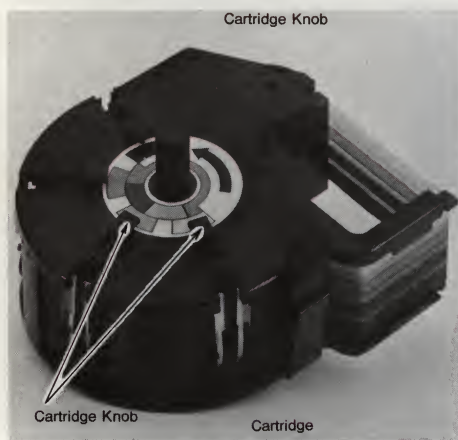


Figure 12.

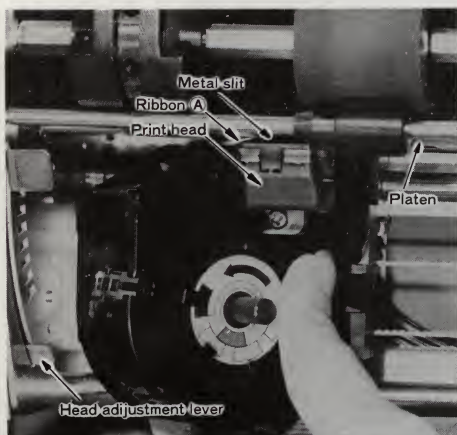


Figure 13.

- (1) Unwrap the ribbon cartridge. Take away the round-shaped foam rubber which is inserted into the cartridge knob. Then turn the inker levers 'ON' by pushing them towards the inside (Refer to Figure 15.).
- (2) Turn the cartridge knob in the direction of the arrow to draw the ribbon taut.
- (3) Move the head adjustment lever away from you. This moves the printhead away from the platen.
- (4) Insert the ribbon cartridge as shown in Figure 13. with the cartridge being held at an angle and put into position from the right side with the (A) portion of the ribbon sliding in between THE PRINTHEAD AND THE METAL SLIT.
- (5) Push on the lower right of the cartridge with the index finger (as shown in Figure 13.). If a 'click' is not heard that the cartridge has not positioned properly turn the cartridge a little more in the direction of the arrow and press again. The cartridge will eventually click into place. After it is in place, turn the cartridge knob again in the direction of the arrow, to draw the ribbon taut.
- (6) Return the head adjustment lever to its original position. (Refer

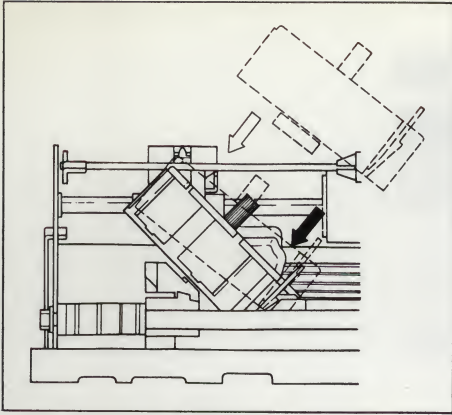


Figure 14.

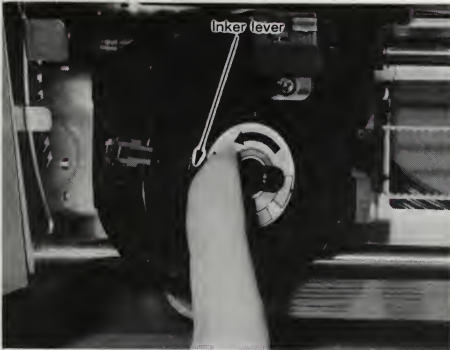


Figure 15.

2. Removal

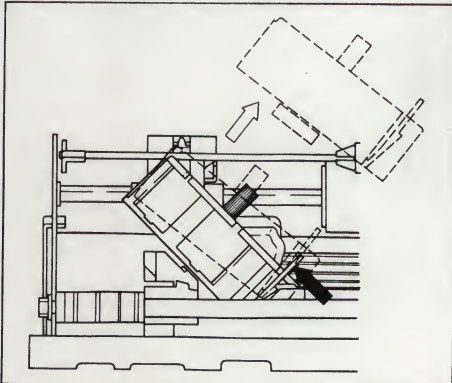


Figure 16.

to [F] HEAD POSITION ADJUSTMENT)

- (7) If a given color is not going to be used, the transfer of ink from that inker to the ribbon can be turned off by pushing that inker lever (Figure 15.) towards the outside of the cartridge.

Pushing the inker lever towards the inside starts the transfer of ink once again.

It is important that the above procedure be used to turn off the ink transfer from unused colors. Failure to do so will cause that portion of the ribbon containing the unused color to keep collecting ink and will eventually result in mixing colors on the ribbon.

(Note) When the ribbon cartridge is shipped out, the inker levers are turedned 'OFF' therefore turn them 'ON' before printing.

- (1) Figure 16. illustrates the removal.
- (2) Remove the cartridge by pressing the black lever towards the cartridge, while firmly gripping part B and simultaneously pull the cartridge off. (Figure16.)

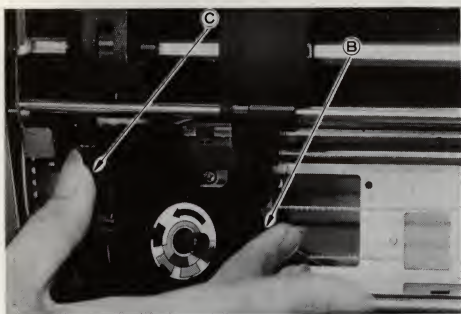


Figure 17.

3. Inker replacement



Figure 18.

Inker may be replaceable once per cartridge on condition that the ribbon is not worn out.

- (1) By slightly pinching the metal clip inwards (part D as in Fig 18.) and pushing the inker lever off.
- (2) Replace the inker by following the reverse order of 1 above, be sure that the color of the ink being inserted matches that of the color indicator label on the top of the cartridge.



Figure 19.

F. HEAD POSITION ADJUSTMENT

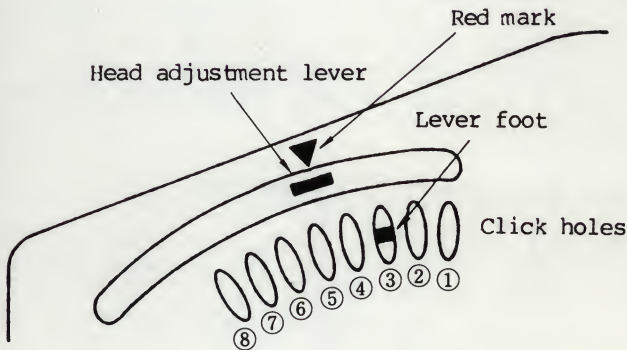
The head adjustment lever is used to adjust for varying thickness of paper and varying printing conditions.

The head adjustment lever is moved by pressing it toward the outside as it is moved back and forth.

When adjusting, make sure that the foot of the lever is set in one of the click holes. The head is adjusted by moving the lever toward (8) for thick paper and toward (1) for thin paper.

The printer is shipped with the lever foot set position (3).

The lever foot should be set to (8) for insertion and removal of the ribbon.



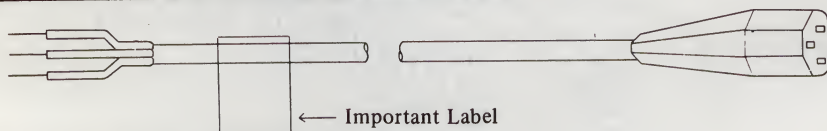
(Fig 20.) Internal left side view

G. CONNECTING THE PRINTER TO YOUR COMPUTER

To connect your printer your COMMODORE computer, please follow the instruction in the order listed below.

1. Make sure that all your equipment is turned OFF, taking the relevant precautions with the diskettes.
2. Connect one socket of the 6-pin DIN socket cable to the connector holes located in the back of your printer at the lower left. This cable is "keyed" so that you cannot plug the cable in the wrong way. This means that the pins should be positioned so that a slight pressure will insert the cable properly. You should not have to force these pins in, doing so would damage the socket.
3. Connect the other socket in the Serial Port Connector either in the back of your computer (see Figure 17.) or in your Disk Drive (see Figure 18.). Again make sure that you are "keying" the pins properly with the connector with 6 holes.
4. Plug into a standard AC wall outlet, but do NOT turn the equipment on yet.

UK RELATIVE MAINS CONNECTION INFORMATION



WARNING: THIS APPARATUS MUST BE EARTHED.

IMPORTANT: The wires in this mains lead are coloured in accordance with the following code.

Green-Yellow: Earth
Blue: Neutral
Brown: Live

As the colours of the wires in the mains lead of this apparatus may not correspond with any coloured markings found identifying the terminals in your plug therefore proceed as follows:

The wire which is colored green-yellow must be connected to the plug terminal which is marked by the letter E, or by the safety earth symbol or coloured green or green-yellow. The longest pin is the earth connector.

The wire which is coloured brown must be connected to the terminal which is marked with the letter L, or coloured Red, or is connected to a fuse. The terminal on the right is the live connector.

The wire which is coloured blue must be connected to the terminal which is marked with the letter N, or coloured Black. The terminal situated on the left is the neutral connector.

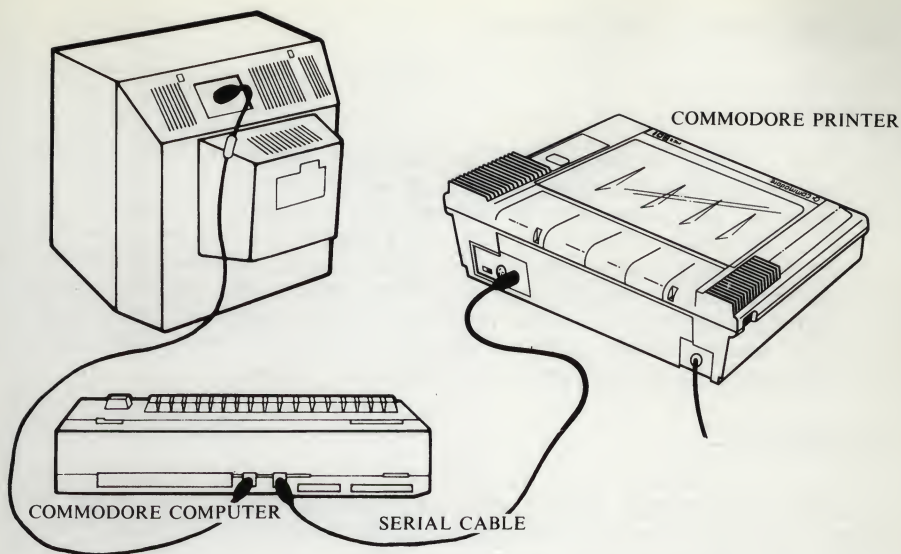


Figure 21. Printer to Computer Hookup

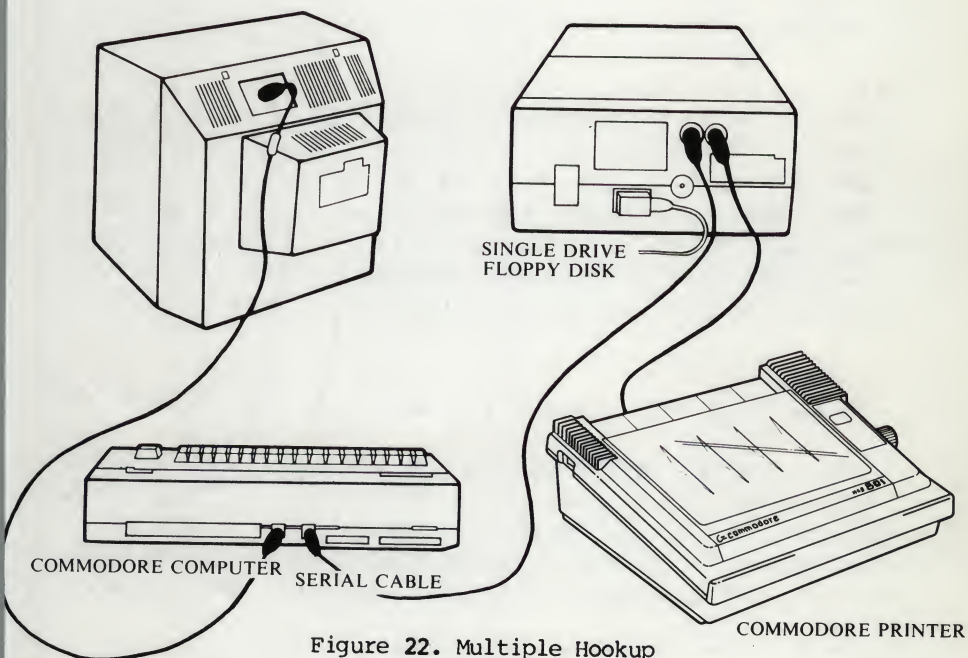


Figure 22. Multiple Hookup

H. PERFORMING THE POWER=ON TEST

You are now ready to proceed with the power-on part of the checkout:

1. Plug in the printer, turn on all your equipment. If a disk drive is connected ensure to turn it on last.
2. Turn on the power to the printer by pressing the switch on the left side of the printer. In response to the application of AC power, the power light should light and the printer's microprocessor should move the print head a few inches to the right, then to it's home position at the left. If this does not happen (and that's highly unlikely), turn off both machines, check contact your COMMODORE dealer.

I. PRINT TEST

Now you can test the print head and the ribbon cartridge. To perform this test, simply turn off the printer, then turn it back on while pressing the PAPER ADVANCE button. It prints the whole of the character spectrum.

NOTE: The PRINT TEST will print 80 characters/line, so use A4 size paper.

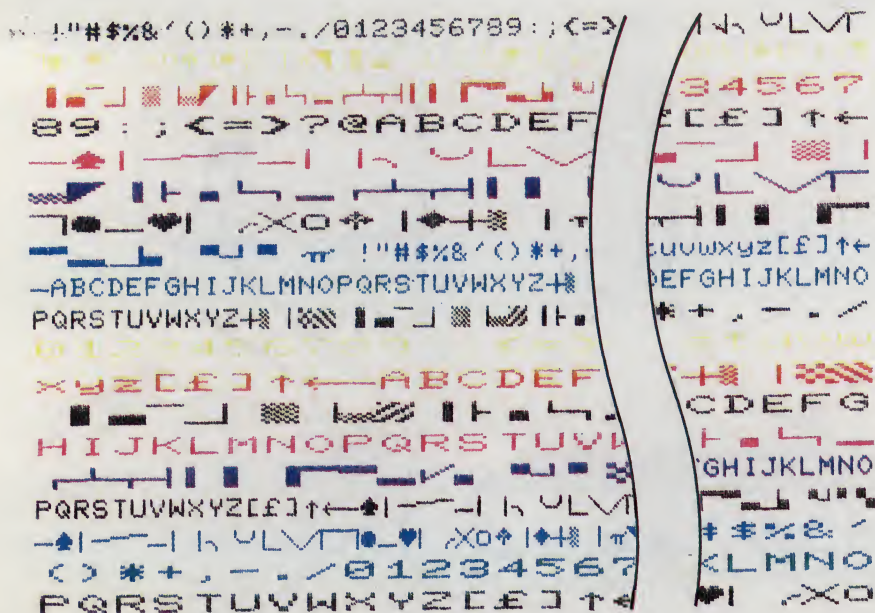


Figure 23. Print test (Character spectrum)

J. TROUBLE SHOOTING

We hope you don't have any problems with your color printer. You do, the table below may help. If you still have problems, return the printer into your COMMODORE dealer.

<u>PROBLEM</u>	<u>PROBABLE CAUSES & SOLUTIONS</u>
Printer won't print.	cause: Printer
Power indicator 'OFF'	solution: Check the power line connection and the power switch.
	cause: Fuse may be blown.
	solution: Replace with a fuse of the same rating (650 MA, 250V)
Printer won't print	cause: Improper connection to computer.
	solution: Check all cable connections.
	cause: Improper loaded cartridge.
	solution: Reload and adjust the cartridge.
Printer okay, paper won't advance.	cause: Paper is jammed.
	solution: Remove and reload the paper.
Printed characters are too light or smudging.	cause: Old or worn out ribbon cartridge.
	solution: Replace the cartridge.
	cause: Wrong cartridge setting.
	solution: Readjust the cartridge.

IMPORTANT: If you try to fix internal problems yourself by working inside your printer, you will void your warranty. Take the printer to your COMMODORE dealer for any necessary repairs.

K. PRECAUTIONS

1. Wait at least two seconds to turn on the power after it is turned off so that the printer will be initialized properly.
2. Never plug in or unplug the serial interface connector while the printer is powered up.
3. Never place the printer in direct sunlight.
4. Never turn the power off while the printer is operating except in the case of #7.
5. Never try to move the print head manually when the power is either on or not.
6. Never stop the moving parts while it is printing.
7. If an object falls into the printer, turn the printer off immediately and remove the object. (This situation is the only exception to item #4 above.)
8. Do not touch the print head while the power is on.

CHAPTER 2

USING YOUR PRINTER

A. INTRODUCTION

Now that you have learned how to thoroughly install your new your new Color Printer, how to print test and how to correct any problems that may occur, you are ready for the next step -- putting the MCS-801 Color Printer to work in your system.

This chapter defines the process of program listings, program results and graphic displays. Before we go any further, make sure that you are able to do the following:

1. Operate your COMMODORE computer.
2. Write elementary programs in BASIC language.
3. OPEN and CLOSE files.
4. Read and write files to various peripheral devices:(like a DATASETTE * recorder or COMMODORE floppy disk drive for the purpose of saving/retrieving programs keyed in as you continue through the rest of this manual.)

If you are not familiar with any of the above items, takes the time now to read and understand USER'S MANUAL and PROGRAMMER'S GUIDE .

B. BASIC COMMANDS ASSOCIATED WITH YOUR PRINTER

1. The OPEN Command:

This command creates a linkage (or correspondence) between a given physical device and a file number. Its format is as follows:

```
OPEN lfn,dn,sa
```

where:

lfn

Is the logical file number (any number from 1 to 255). If the logical file number is greater than 127, any printed lines will be double spaced.

dn

Is the device number of the peripheral that will be sending/receiving the file. For the remainder of this manual, you will be using #4, the address of your printer. (Since that is the number assigned to it at the factory.)

sa

Is the secondary address, a unique way of instructing your printer's internal microprocessor to perform certain specific tasks. More about this important feature later.

2. The PRINT# Command:

This command works just like the BASIC PRINT command except that it directs the output to the printer instead of the video screen. The abbreviation for PRINT# is pR (unshifted p and shifted R). There must be no space before or after the #. The format is as follows:

```
PRINT# lfn <,variable/data>
```

where:

lfn

Is the logical file number (any number from 1 to 255). This number must correspond to a given OPEN command.

<variable/data> will be discussed in detail later in this user's guide.

3. The CLOSE Command:

Careful use of this command is important because you may only have a maximum of ten files open at any one time. Programming your printer may require that you keep several files open simultaneously. Always close files when you are finished using them. The format is as follows:

```
CLOSE lfn
```


where:

lfn

Is the logical file number of the file being closed. This number must correspond to a given OPEN command.

4. The CMD Command:

Good programming practice dictates never using this statement within a program; use this statement in direct mode only.

This statement transfers the primary output device(TV screen) to the logical file number specified in the command. The file must have been previously OPENed. When this command is in effect, all output generated by PRINT or LIST command will be sent to the file instead of the video monitor. The format of the command is as follows:

CMD lfn<,variable/data>

where:

lfn

Is the logical file number to receive the output.

<variable/data> (optional) when specified is sent directly to the file. This can be used as a handy way of printing titles on printouts.

To re-direct the output back to the primary output device, the PRINT# (not abbreviation) command must be used to send a blank line to the CMD device prior to closing the file. If a SYNTAX ERROR occurs, output will not be redirected back to the video monitor. Devices are not "unlistened" by this so you should PRINT# a blank line after an error condition.

5. Command Example

OPEN 1,4,7	File#1, device#4, secondary address#7
OPEN 2,4	File#2, device#4 (secondary address is default #0)
PRINT#1,"HELLO THERE"	Print "HELLO THERE" on file#1
PRINT#2,""	Print a blank line on file#2
CLOSE1:CLOSE2	Close file#1 and file#2

Remember, since CMD does not close the line to the printer, you must always precede a CLOSE command with PRINT# in order to properly close the file.

EXAMPLE:

Right

Wrong

OPEN 1,4
PRINT#1,"HELLO THERE"
CLOSE 1

OPEN 2,4,7
CMD 2,"HELLO THERE" not
PRINT#2
CLOSE 2

OPEN 3,4
CMD 3,"HELLO THERE" not
PRINT#3,"HELLO THERE"
CLOSE 3

OPEN 4,4,7
PRINT#4,"HELLO THERE" not
CMD 4,"HELLO THERE"
PRINT#4
CLOSE 4

OPEN 2,4,7
CMD 2,"HELLO THERE"
CLOSE 2

OPEN 3,4
CMD 3,"HELLO THERE"
PRINT#3,"HELLO THERE"
PRINT#3: CLOSE3

OPEN 4,4,7
PRINT#4,"HELLO THERE"
CMD 4,"HELLO THERE"
CLOSE 4

Once you are aware of what you can and you cannot do, you can proceed to the next part of this chapter, which describes how to use these commands to control your printer.

C. PRINTING IN THE DIRECT MODE

Having been acquainted with the BASIC commands, you will now need to handle files directed to your printer, let's put these commands to use in some practical applications. The direct mode allows you to communicate with your printer by entering printing commands directly into your computer via the keyboard and having them executed immediately.

The following example illustrates the DIRECT mode procedure for listing a short BASIC program. In the example below, a single statement BASIC program is keyed into the computer's memory. A LIST command is keyed and the program is printed on your video monitor. A file(#3) is OPENed to your Printer. A CMD3 is issued to the file open on your printer(it is now listening). Issue the LIST command again. This time the output will be sent to your printer. To break the CMD command you instruct the computer to print a null character to file#3. Now you can CLOSE file#3, this file number can be used for some other use now all output will now appear on the video monitor.

You Type:	The screen displays:	The Printer prints:
-----------	----------------------	---------------------

10 PRINT"TEST"	10 PRINT"TEST"	
----------------	----------------	--

LIST	LIST	
	10 PRINT"TEST"	

OPEN 3,4	OPEN 3,4	
	READY.	

CMD3	CMD3	READY.
------	------	--------

LIST	LIST	10 PRINT"TEST"
		READY.

PRINT#3	PRINT#3	
	READY.	

CLOSE3	CLOSE3	
	READY.	

1. Program List

Program LISTing is the most important job of the printer. To get a program list, follow the procedure below.

1. Program List in Graphic Mode

OPEN 4,4:CMD4:LIST	Simple Listing
OPEN 4,4:CMD4:LIST 100-500	Simple Listing Line No.100-500
OPEN 4,4:CMD4,"PROGRAM NAME":LIST	Print the Program Name and Listing
OPEN 4,4:CMD4,CHR\$(14)"PROGRAM NAME":LIST	Print the Program Name in enhance character and Listing
OPEN 4,4:CMD4,CHR\$(147)"PROGRAM NAME":LIST	Listing on Paging Mode
OPEN 4,4:CMD4,CHR\$(147)CHR\$(14)"PROGRAM NAME":LIST	Print the Program Name in enhance character and Listing in Paging Mode.

After getting the program list, enter the following.

PRINT#4
CLOSE4

or
PRINT#4,CHR\$(12).....This performs a Form Feed
CLOSE4

Note: Control Character CHR\$(14),CHR\$(147) are used in above procejure. You may find out more about these on Chapter2 Part F (Also see your USER'S MANUAL for more detail about keyboard control characters.)

2. To get Program List in Business Mode, You should add the secondary address of 7 to the OPEN Statement.

OPEN 4,4,7:CMD4:LIST	Simple Listing
OPEN 4,4,7:CMD4:LIST 100-500	Simple Listing Line No.100-500
OPEN 4,4,7:CMD4,"PROGRAM NAME":LIST	Print the Program Name and Listing
OPEN 4,4,7:CMD4,CHR\$(14)"PROGRAM NAME":LIST	Print the Program

Name in enhance
character and
Listing

OPEN 4,4,7:CMD4,CHR\$(147)"PROGRAM NAME":LIST

Listing in Paging
Mode

OPEN 4,4,7:CMD4,CHR\$(147)CHR\$(14)"PROGRAM NAME":LIST

Print the program
name in enhance
character and
Listing in Paging
Mode

After getting the PROGRAM LIST, enter the following

PRINT#4
CLOSE4

or

PRINT#4,CHR\$(12).....This performs a Form Feed
CLOSE4

D. PRINTING UNDER PROGRAM CONTROL

You have seen how to print a list of a simple BASIC program directly from memory. The following program example illustrates how to print a BASIC program list under the control of the program itself, if you look at the program below, you may have noted that CMD and LIST are in it, this, as you are now probably aware, is generally not good programming practice.

```
10 OPEN 3,4
20 CMD 3
30 PRINT"PROGRAM CONTROL"
40 LIST
```

If you typed List, the program you just entered would be displayed on your video monitor. Enter the RUN command. The following should be printed on your printer:

PROGRAM CONTROL

```
10 OPEN 3,4
20 CMD3
30 PRINT"PROGRAM CONTROL"
40 LIST
```

READY.

Now, enter the following.

```
PRINT#3
CLOSE3
```

The RUN command instructed the computer to execute the program in its memory. First, file#3 (device#4) was opened. Next, the CMD command instructed the computer to direct all subsequent output to file#3. The program then PRINTed PROGRAM CONTROL on file #3. The LIST command instructed the computer a listing of the program in its memory on file#3.

REMEMBER --When using the LIST command within a program, you must type the PRINT# lfn command to break the CMD command over the printer. Then you must type the CLOSE lfn command. Using the LIST command or the CMD command in a program is generally not good programming practice. It is suggested that you only use them in DIRECT mode.

E. HOW TO PRINT, HOW TO SPECIFY COLOR

There are three types of printing mode with this color printer.

- a. Text (character) printing
- b. Special character design printing
- c. Scan mode bit image printing

And, there are two types of color specification.

a. DC4 CHR\$(n) specifies color for text and special character design printing.

b. Scan mode bit image printing includes dot color specification.

1. Text printing and color specification

Text printing is done by following sequence.

Example1

```
100 REM TEXT PRINTING
110 OPEN 4,4 :REM PRINT IN GRAPHIC MODE
120 PRINT#4,"TEXT PRINTING"
130 FOR I=1 TO 15:PRINT#4:NEXT I
140 CLOSE4
```

RUN

TEXT PRINTING

When power is turned on or when the secondary address of 10 is send or when the RESET signal is input, the color is set to black.

COLOR SPECIFICATION

a) Color specification is done by DC4 code [CHR\$(20)] and ASCII CHR\$(n) (0<=n<=7).

DC4 CHR\$(n) specifies any of 8 colors.

b) n is a decimal number where 0<=n<=7. The following colors are selected according to the value of n.

n = 0	black
n = 1	white (non print, paper white)
n = 2	red
n = 3	cyan (light blue)
n = 4	purple
n = 5	green

n = 6 blue
n = 7 yellow

Example 2

```
100 REM DC4[CHR$(20)]...SPECIFY COLOR
110 OPEN 4,4,0:REM PRINT IN GRAPHIC MODE
120 DC$=CHR$(20)
130 PRINT#4,DC$;CHR$(2);:REM RED
140 PRINT#4,"AB";
150 PRINT#4,DC$;CHR$(3);:REM CYAN
160 PRINT#4,"CD";
170 FOR I=1 TO 15:PRINT#4:NEXT I
180 CLOSE4
```

RUN

ABCD

c) The color remains unchanged until the next DC4 CHR\$(n) is input.

Example 3

```
100 REM DC4[CHR$(20)]...SPECIFY COLOR
110 OPEN 4,4,0:REM PRINT IN GRAPHIC MODE
120 DC$=CHR$(20)
140 PRINT#4,"COLOR REMAINS"
170 FOR I=1 TO 15:PRINT#4:NEXT I
180 CLOSE4
```

RUN

Some more example programs are shown below.

Example 4

```
100 REM DC4[CHR$(20)]...SPECIFY COLOR
110 OPEN 4,4,0:REM PRINT IN GRAPHIC MODE
120 DC$=CHR$(20)
130 A$="MCS-881 COLOR PRINTER"
140 FOR I=0 TO 7
150 PRINT#4,DC$;CHR$(I);
160 PRINT#4,A$
170 NEXT I
180 FOR J=1 TO 15:PRINT#4:NEXT J
```

190 CLOSE4

RUN

MCS-801 COLOR PRINTER

MCS-801 COLOR PRINTER

MCS-801 COLOR PRINTER

MCS-801 COLOR PRINTER

MCS-801 COLOR PRINTER

MCS-801 COLOR PRINTER

MCS-801 COLOR PRINTER

Example 5

```
100 REM DC4[CHR$(20)]...SPECIFY COLOR
102 POKE 53281,1 :REM C64
104 POKE 36879,PEEK(36879)AND15 :REM VIC-20
110 OPEN 4,4,0:REM PRINT IN GRAPHIC MODE
120 DC$=CHR$(20)
125 COLR$="■ ■ ■ ■ ■ ■ ■ ■"
130 A$="MCS-801 COLOR PRINTER"
140 FOR I=0 TO 7
150 PRINT#4,DC$;CHR$(I);
155 PRINT MID$(COLR$,I+1,1);
160 PRINT#4,A$
165 PRINT A$
170 NEXT I
180 FOR J=1 TO 15:PRINT#4:NEXT J
190 CLOSE4
```

RUN

MCS-801 COLOR PRINTER

MCS-801 COLOR PRINTER

MCS-801 COLOR PRINTER

MCS-801 COLOR PRINTER

MCS-801 COLOR PRINTER

MCS-801 COLOR PRINTER

MCS-801 COLOR PRINTER

Example 6

```
100 REM DC$(CHR$(20))...SPECIFY COLOR
110 OPEN 4,4 :REM PRINT IN GRAPHIC MODE
120 DC$=CHR$(20)
140 FOR I=0 TO 7
145 READ A$(I)
150 PRINT#4,DC$;CHR$(I);
160 PRINT#4,A$(I)
170 NEXT I
180 FOR J=1 TO 15:PRINT#4:NEXT J
190 CLOSE4
200 DATA BLACK,WHITE,RED,CYAN
210 DATA PURPLE,GREEN,BLUE,YELLOW
```

RUN

BLACK

RED

CYAN

PURPLE

GREEN

BLUE

YELLOW

example 7

```
100 REM DC$(CHR$(20))...SPECIFY COLOR
102 FOR I=1 TO 6 :READ C(I):NEXT I
110 OPEN 4,4 :REM PRINT IN GRAPHIC MODE
120 DC$=CHR$(20)
130 A$="MCS-801 COLOR PRINTER"
140 FOR I=1 TO LEN(A$)
150 PRINT#4,DC$;CHR$(C(I)AND7));
160 PRINT#4,MID$(A$,I,1);
170 NEXT I
175 PRINT#4
180 FOR J=1 TO 15:PRINT#4:NEXT J
190 CLOSE4
200 DATA 0,2,3,4,5,6
```

RUN

MCS-801 COLOR PRINTER

2. Special Character Printing

ESC K n3n2n1 graphic data

ESC K specifies data for special character design; n3 n2 n1 specifies the desired number of dot columns. Then the same number of graphic data will be input.

n3 n2 n1 is input using the ASCII codes to represent three decimal numbers (001 through 640). When a number greater than 640 is specified, it is read as 640. When 000 is specified, the first 5 bytes of this command, that is, ESC K n3 n2 n1 are ignored.

The 1st printed dot corresponds to D1 of the graphic data; the 8 th dot correspondence to D8.

Example

1●.....1st dot
2●
4○
8●
16○
32○
64●
128○.....8th dot

Input data of the above is as follows. In this example, the amount of data is \$4B = 75.

D8	D7	D6	D5	D4	D3	D2	D1
0	1	0	0	1	0	1	1
↑						↑	
the 8th dot						the 1st dot	

DC4 CHR\$(n) specifies any of 8 colors

n is a number where $0 \leq n \leq 7$. The following colors are selected according to the value of n.

n = 0 black
n = 1 white (non print, paper white)
n = 2 red
n = 3 cyan
n = 4 purple
n = 5 green
n = 6 blue
n = 7 yellow

The colors remains unchanged until the DC4 CHR\$(n) is input.

When power is turned on or when the secondary address of 10 is sent or when the RESET signal is input, the color is set to black.

Example 1:

Lets try printing the following bit-figure.

```

1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0
2 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0
4 0 0 1 1 1 1 1 1 1 1 1 0 0 0 0
8 0 0 1 1 0 0 1 1 0 0 1 1 0 0 0 0
16 1 1 1 1 0 0 1 1 0 0 1 1 1 1 0 0
32 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0
64 0 0 1 1 0 0 1 1 0 0 1 1 0 0 0 0
128 0 0 1 1 0 0 1 1 0 0 1 1 0 0 0 0

```

Program:

```

100 REM 8-BIT GRAPHIC DATA PRINTING
110 REM ESC;"K";"NNN";CHR$(GRAPHIC DATA)
120 OPEN 4,4 :REM PRINT IN GRAPHIC MODE
130 DC$ =CHR$(20)
140 ESC$=CHR$(27)
150 :
160 A$=""
170 FOR I=0 TO 15
180 READ A: A$=A$+CHR$(A)
190 NEXT I
200 FOR J=0 TO 7
210 IFJ=10RJ=7THEN240:REM AVOID WHITE/YELLOW
220 PRINT#4,DC$;CHR$(J);: REM SET COLOR
230 PRINT#4,ESC$;"K";"016";A$
240 NEXT J
250 :
260 FOR K=1 TO 15:PRINT#4:NEXT K
270 CLOSE4
280 END
290 :
300 DATA 48,48,252,252,38,38,255,255
310 DATA 38,38,252,252,48,48,0,0

```

READY.



Exercise:

Now try to make the following figure in above method.

1	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0
2	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0
4	0	0	1	1	1	1	0	0	1	1	1	1	0	0	0
8	1	1	1	1	0	0	0	0	0	0	1	1	1	1	0
16	0	0	1	1	1	1	0	0	1	1	1	1	0	0	0
32	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0
64	0	0	1	1	0	0	1	1	0	0	1	1	0	0	0
128	1	1	0	0	0	0	1	1	0	0	0	0	1	1	0

Example 2:

Data is sent to the print head and is printed bit-by-bit (dot-by-dot). Each printed byte forms the basis for the characters image printed on a larger scale than the previous programs. The following program will print a graphic image created by several bytes joined together.

```
100 REM 8-BIT GRAPHIC DATA PRINTING
110 REM ESC;"K";"NNN";CHR$(GRAPHIC DATA)
120 OPEN 4,4 :REM PRINT IN GRAPHIC MODE
130 DC$ =CHR$(20)
140 ESC$=CHR$(27)
150 :
160 A$="" :FOR I=1 TO 255 :A$=A$+CHR$(I) :NEXT I
170 B$=" ALL BIT PATTERN 1"
180 :
190 C$="" :FOR J=255 TO 1 STEP -1 :C$=C$+CHR$(J) :NEXT J
200 D$=" ALL BIT PATTERN 2"
210 :
220 FOR K=0 TO 7
230 PRINT#4,DC$;CHR$(K);
240 PRINT#4,ESC$;"K";"255";A$;
250 PRINT#4,DC$;CHR$((K+2)AND 7);
260 PRINT#4,B$
270 NEXT K
280 PRINT#4
290 :
300 FOR L=7 TO 0 STEP -1
310 PRINT#4,DC$;CHR$(L);
320 PRINT#4,ESC$;"K";"255";C$;
330 PRINT#4,DC$;CHR$((L+2)AND 7);
335 PRINT#4,D$
340 NEXT L
350 PRINT#4
```

```

360 :
370 FOR M=1 TO 15:PRINT#4:NEXT M
380 CLOSE4
390 END

```

RUN



```

ALL BIT PATTERN
ALL BIT PATTERN
ALL BIT PATTERN
ALL BIT PATTERN
ALL BIT PATTERN
ALL BIT PATTERN
ALL BIT PATTERN
ALL BIT PATTERN
ALL BIT PATTERN
ALL BIT PATTERN
ALL BIT PATTERN
ALL BIT PATTERN
ALL BIT PATTERN
ALL BIT PATTERN
ALL BIT PATTERN

```

Mixed character printing and special character design printing can exist in same line.

Example:

DC4 CHR\$(2)	Character	text	DC4 CHR\$(3)	Character	CR
	data 1	data 1		data 2	

Character data1 and text data 1 will be printed in the color red and character data 2 will be printed in the color cyan. In this way the colors for mixed text and graphic images may be specified arbitrarily.

Example 3:

```

100 REM MIXED PRINTING
110 OPEN 4,4 :REM PRINT IN GRAPHIC MODE
120 DC$ =CHR$(28)
130 ESC$=CHR$(27)
140 :
150 FOR I=1 TO 8:READ A:A$=A$+CHR$(A):NEXT I

```

```

160 FOR J=1 TO 8:READ B:B$=B$+CHR$(B):NEXT J
170 :
180 PRINT#4,DC$;CHR$(2);      :REM RED
190 PRINT#4,ESC$;"K";"008";A$;:REM GRAPHIC 1
200 PRINT#4," COMMODORE LOGO ";
210 PRINT#4,DC$;CHR$(3);      :REM CYAN
220 PRINT#4,ESC$;"K";"008";B$;:REM GRAPHIC 2
230 :
240 FOR K=1 TO 15:PRINT#4:NEXT K
250 CLOSE4
260 END
270 :
280 DATA 56,68,130,130,108,68,0,0
290 DATA 0,0,68,108,130,130,68,56

```

RUN

6 COMMODORE LOGO 3

Specifying different colors may be done any number of times within a given row and the colors may be used in any order.

Example 4:

1st dot-----2 7 7	color code
0 6 1	0 ... black
1 2 5	1 ... white (non print)
1 4 6	2 ... red
5 1 0	3 ... cyan
4 5 1	4 ... purple
3 1 2	5 ... green
8th dot-----1 3 1	6 ... blue
^ ^ ^	7 ... yellow

dot position 0 ————↑

dot position 1 ————↑

dot position 2 ————↑

Example 3 shows the colors in any given column can be mixed in any order. Input for this pattern in 8-dot vertical columns is as follows. Data with the \$ sign represents hexadecimal data.

DC4 \$00	ESC POS "00"	ESC K "003"	\$02 \$04 \$40
set black	start at dot position 0	input 3 columns of dots	dots to be printed in black

DC4 \$01 set white	ESC POS "00"	ESC K "003"	\$8C \$50 \$A2 dots to be printed in white
DC4 \$02 set red	ESC POS "00"	ESC K "003"	\$01 \$04 \$40 dots to be printed in red
DC4 \$03 set cyan	ESC POS "00"	ESC K "002" input 2 columns of dots	\$40 \$80 dots to be printed in cyan
DC4 \$05 set green	ESC POS "00"	ESC K "003"	\$10 \$20 \$04 dots to be printed in green
DC4 \$06 set blue	ESC POS "01"	ESC K "002"	\$02 \$08 dots to be printed in blue
DC4 \$03 set yellow	ESC POS "01"	ESC K "002"	\$40 \$80 dots to be printed in yellow

In this way, the printed data for a given position can be placed on top of other data any number of times using the position command and specifications can be repeated any number of times.

Program

```

100 REM MULTIPLE COLOR DOT SPECIFICATION
110 REM ESC;"K";"NNN"
120 REM ESC;POS;CHR$(NH)CHR$(NL)
130 OPEN 4,4 :REM PRINT IN GRAPHIC MODE
140 DC$ =CHR$(20)
150 ESC$=CHR$(27):PO$=CHR$(16)
160 :
170 PRINT#4,DC$;CHR$(0); :REM BLACK
172 PRINT#4,ESC$;PO$;CHR$(0)CHR$(0);
174 PRINT#4,ESC$;"K";"003";CHR$(4)CHR$(4)CHR$(64)
180 PRINT#4,DC$;CHR$(1); :REM WHITE
182 PRINT#4,ESC$;PO$;CHR$(0)CHR$(0);
184 PRINT#4,ESC$;"K";"003";CHR$(140)CHR$(80)CHR$(162);
190 PRINT#4,DC$;CHR$(2); :REM RED

```

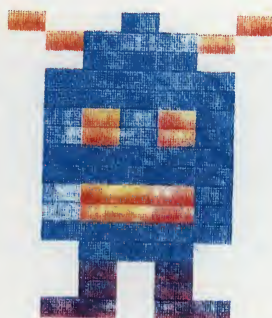
```

192 PRINT#4,ESC$;PO$;CHR$(0)CHR$(0);
194 PRINT#4,ESC$;"K";"003";CHR$(1)CHR$(4)CHR$(64);
200 PRINT#4,DC$;CHR$(3);          :REM CYAN
202 PRINT#4,ESC$;PO$;CHR$(0)CHR$(0);
204 PRINT#4,ESC$;"K";"002";CHR$(64)CHR$(128);
210 PRINT#4,DC$;CHR$(4);          :REM PURPLE
212 PRINT#4,ESC$;PO$;CHR$(0)CHR$(0);
214 PRINT#4,ESC$;"K";"002";CHR$(32)CHR$(8);
220 PRINT#4,DC$;CHR$(5);          :REM GREEN
222 PRINT#4,ESC$;PO$;CHR$(0)CHR$(0);
224 PRINT#4,ESC$;"K";"003";CHR$(16)CHR$(32)CHR$(4);
230 PRINT#4,DC$;CHR$(6);          :REM BLUE
232 PRINT#4,ESC$;PO$;CHR$(0)CHR$(1);
234 PRINT#4,ESC$;"K";"002";CHR$(2)CHR$(8);
240 PRINT#4,DC$;CHR$(7);          :REM YELLOW
242 PRINT#4,ESC$;PO$;CHR$(0)CHR$(1);
244 PRINT#4,ESC$;"K";"002";CHR$(1)CHR$(1);
250 PRINT#4
260 :
270 FOR I=1 TO 15:PRINT#4:NEXT I
280 CLOSE4

```

Example 5:

Try to print the following figure



```

1000 REM MULTIPLE COLOR DOT SPECIFICATION
1010 REM ESC;"K";"NNN"
1020 REM ESC;POS;CHR$(NH)CHR$(NL)
1030 DIM B(8),MC(3)
1040 B(0)=0:B(1)=1:B(2)=2:B(3)=4:B(4)=8
1050 B(5)=16:B(6)=32:B(7)=64:B(8)=128
1060 MC(0)=0:MC(1)=2:MC(2)=3:MC(3)=4
1070 OPEN#4,4,0 :REM PRINT IN GRAPHIC MODE
1080 OPEN#6,4,6
1090 PRINT#6,CHR$(13):REM SET LINE SPACING
1100 DC$=CHR$(20)
1110 ESC$=CHR$(27):PO$=CHR$(16)
1120 :
1130 N=9:GOSUB1230:
1140 N=9+16:GOSUB1230:PRINT#4,CHR$(10);
1150 N=9:GOSUB1230
1160 N=9+16:GOSUB1230:PRINT#4,CHR$(10);
1170 :
1180 PRINT#6,CHR$(24):REM RESET ORG. SPACING
1190 FOR I=1 TO 15:PRINT#4:NEXT I
1200 CLOSE#4
1210 END
1220 :
1230 FOR ROW=1 TO 8
1240 FOR MC=0 TO 3
1250 READ Q
1260 FOR Z= 3 TO 0 STEP-1
1270 M$(ROW,MC,Z)=""
1280 READ O :IF O=0 THEN B=B(0)
1290 :IF O=1 THEN B=B(ROW)
1300 FOR I =1TO4
1310 M$(ROW,MC,Z)=M$(ROW,MC,Z)+CHR$(B)
1320 NEXT I
1330 NEXT Z
1340 :
1350 C=Q:GOSUB1420
1360 PRINT#4,M$(ROW,MC,3);M$(ROW,MC,2);M$(ROW,MC
1);M$(ROW,MC,0);
1370 :
1380 NEXT MC
1390 NEXT ROW
1400 RETURN
1410 :
1420 PRINT#4,DC$;CHR$(C);
1430 PRINT#4,ESC$;PO$;CHR$(0)CHR$(N);
1440 PRINT#4,ESC$;"K";"016";

```


1450 RETURN

1460 :

1470 DATA 1,0,1,1,0,2,1,0,0,0,1,0,0,0,0,3,0,0,0,1
1480 DATA 1,1,0,0,0,2,0,1,0,0,1,0,0,0,0,3,0,0,1,1
1490 DATA 1,1,1,0,0,2,0,0,0,0,1,0,0,0,0,3,0,0,1,1
1500 DATA 1,1,0,0,0,2,0,0,0,0,1,0,0,0,0,3,0,1,1,1
1510 DATA 1,1,0,0,0,2,0,0,0,0,1,0,0,0,0,3,0,1,1,1
1520 DATA 1,1,0,0,0,2,0,0,1,0,1,0,0,0,0,3,0,1,0,1
1530 DATA 1,1,0,0,0,2,0,0,1,0,1,0,0,0,0,3,0,1,0,1
1540 DATA 1,1,0,0,0,2,0,0,0,0,1,0,0,0,0,3,0,1,1,1
1550 DATA 1,1,1,0,1,2,0,0,1,0,1,0,0,0,0,3,0,0,0,0
1560 DATA 1,0,0,1,1,2,0,1,0,0,1,0,0,0,0,3,1,0,0,0
1570 DATA 1,0,1,1,1,2,0,0,0,0,1,0,0,0,0,3,1,0,0,0
1580 DATA 1,0,0,1,1,2,0,0,0,0,1,0,0,0,0,3,1,1,0,0
1590 DATA 1,0,0,1,1,2,0,0,0,0,1,0,0,0,0,3,1,1,0,0
1600 DATA 1,0,0,1,1,2,1,0,0,0,1,0,0,0,0,3,0,1,0,0
1610 DATA 1,0,0,1,1,2,1,0,0,0,1,0,0,0,0,3,0,1,0,0
1620 DATA 1,0,0,1,1,2,0,0,0,0,1,0,0,0,0,3,1,1,0,0
1630 DATA 1,1,0,0,0,2,0,0,0,0,1,0,0,0,0,3,0,1,1,1
1640 DATA 1,1,0,0,0,2,0,0,1,1,1,0,0,0,0,3,0,1,0,0
1650 DATA 1,1,0,0,0,2,0,0,1,1,1,0,0,0,0,3,0,1,0,0
1660 DATA 1,1,0,0,0,2,0,0,0,0,1,0,0,0,0,3,0,1,1,1
1670 DATA 1,1,1,0,0,2,0,0,0,0,1,0,0,0,0,3,0,0,1,1
1680 DATA 1,1,1,0,1,2,0,0,0,0,6,0,0,1,0,3,0,0,0,0
1690 DATA 1,1,1,0,1,2,0,0,0,0,6,0,0,1,0,3,0,0,0,0
1700 DATA 1,1,0,0,1,2,0,0,0,0,6,0,1,1,0,3,0,0,0,0
1710 DATA 1,0,0,1,1,2,0,0,0,0,1,0,0,0,0,3,1,1,0,0
1720 DATA 1,0,0,1,1,2,1,0,0,0,1,0,0,0,0,3,0,1,0,0
1730 DATA 1,0,0,1,1,2,1,0,0,0,1,0,0,0,0,3,0,1,0,0
1740 DATA 1,0,0,1,1,2,0,0,0,0,1,0,0,0,0,3,1,1,0,0
1750 DATA 1,0,1,1,1,2,0,0,0,0,1,0,0,0,0,3,1,0,0,0
1760 DATA 1,0,1,1,1,2,0,0,0,0,6,1,0,0,0,3,0,0,0,0
1770 DATA 1,0,1,1,1,2,0,0,0,0,6,1,0,0,0,3,0,0,0,0
1780 DATA 1,0,0,1,1,2,0,0,0,0,6,1,1,0,0,3,0,0,0,0

RUN



3. Scan mode bit image printing

This is a means of printing, a screen display created by high-resolution (hi-res) graphics. Each color corresponds to each of the four hammers that comprise the printhead. As each pixel is displayed in it's given colr each bit of the printhead hammer is programmed with the same color and will therefore print using it.

The sequence of the scan mode bit image printing is follows:

ESC C n2nl n0 m2ml m0 color data.....

Data for the dots is input in the order 1 2 3 ...as shown below.

N = n2 nl n0 dots (640 dot maximum)

	1	2	3	N
	N+1				2N
	2N+1				3N
	0				
	0				
	0				
	0				
8th dot ---	0				
1st dot of--	0				
next line	0				
	0				
	0				
	0				
	0				
	0				
	0				

M = m2 ml m0 rows

direction of printing -->

a) ESC C specifies the mode followed by n2 nl n0 and m2 ml m0 which specify the number of dots to be printed in the horizontal direction and the number of dots to be printed in the vertical direction. This is then followed by N x M bytes of data. (N= n2 nl n0 and M = m2 ml m0; Both numbers are input ascii codes to represent these decimal numbers). Both N and M must be within the range 001 to 640 and 3-digit decimal numbers.

b) Each dot is represent by one byte with the color being selected as shown below.

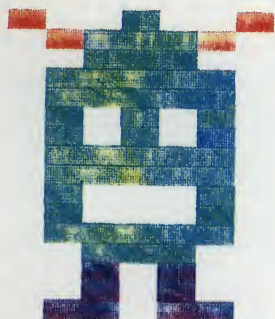
0	...	black
1	...	white (non print, paper white)
2	...	red
3	...	cyan
4	...	purple
5	...	green
6	...	blue
7	...	yellow

c) In this mode the printer starts printing without reading a print command. When "M" is equal to or smaller than 8, printing starts right after data for M times the number of dots are input. When "M" is greater than 8, printing starts right after data for 8 times the number of dots (one line of data for the printer).

d) After printing all the data in the buffer, the printer performs a form feed after which this mode is cancelled automatically.

Example 1:

Let's try to make color printing for the following figure by scan mode bit image printing.



```
100 REM SCAN MODE BIT IMAGE PRINTING
110 REM ESC;"C";"NNN";"MMM";COLOR DATA...
120 OPEN 4,4 :REM PRINT IN GRAPHIC MODE
130 ESC$=CHR$(27)
140 :
150 PRINT#4,ESC$;"C";"016";"016"
160 FOR I=1 TO 16*16
170 READ A:A$=CHR$(A)
180 PRINT#4,A$;
190 NEXT I
200 :
210 FOR J=1 TO 15:PRINT#4:NEXT J
220 CLOSE4
230 END
```

240 :

```
250 DATA 2,2,1,1,1,1,5,5,1,1,1,1,2,2,1,1
260 DATA 1,1,2,2,5,5,5,5,5,5,2,2,1,1,1,1
270 DATA 1,1,1,1,5,5,5,5,5,5,1,1,1,1,1,1
280 DATA 1,1,5,5,5,5,5,5,5,5,5,5,1,1,1,1
290 DATA 1,1,5,5,5,5,5,5,5,5,5,5,1,1,1,1
300 DATA 1,1,5,5,1,1,5,5,1,1,5,5,1,1,1,1
310 DATA 1,1,5,5,1,1,5,5,1,1,5,5,1,1,1,1
320 DATA 1,1,5,5,5,5,5,5,5,5,5,5,1,1,1,1
330 DATA 1,1,5,5,5,5,5,5,5,5,5,5,1,1,1,1
340 DATA 1,1,5,5,1,1,1,1,1,1,5,5,1,1,1,1
350 DATA 1,1,5,5,1,1,1,1,1,1,5,5,1,1,1,1
360 DATA 1,1,5,5,5,5,5,5,5,5,5,5,1,1,1,1
370 DATA 1,1,1,1,5,5,5,5,5,5,1,1,1,1,1,1
380 DATA 1,1,1,1,2,2,1,1,2,2,1,1,1,1,1,1
390 DATA 1,1,1,1,2,2,1,1,2,2,1,1,1,1,1,1
400 DATA 1,1,2,2,2,2,1,1,2,2,2,2,1,1,1,1
```

RUN



Scan mode is mostly used to print high-resolution (hi-res) graphics. Hi-res graphics requires a high degree of data manipulation and BASIC is too slow for this. Therefore programs written with hi-res graphics are generally created using assembly language with this printer.

Note: Color Overstriking Sequence

The color ribbon is designed to print denser color tones on top of lighter tones in a single pass print. The color ribbon has four colored strips and is assembled and struck by the printhead in the following order - yellow, cyan, purple and black. Therefore the printer will saturate the paper with a dot of yellow ink and will then print a dot of purple ink on top of yellow dot forming another tone, if purple was printed first the yellow would make no color change because it is less denser.

E. ADVANCED PROGRAMMING TECHNIQUES THROUGH SECONDARY ADDRESSING

You have previously been introduced about the secondary address concept in Section B containing the OPEN command. This unique feature is used to instruct your new printer to perform the following operations.

<u>Operations</u>	<u>Secondary Address</u>
Print in Graphic Character Mode	0 (default)
Setting the Number of Lines Per Page	3
Defining a Programmable Character	5
Setting Spacing Between Lines	6
Print in Business Character Mode	7
Resetting the Printer	10
Enable 2-pass Mode in Scan Mode Printing	19
Disable 2-pass Mode in Scan Mode Printing	20

For each desired operation you will be OPENing a file and issuing PRINT# commands to that file to instruct your printer with the function defined by the secondary address. For example, a PRINT# command issued to a file that was OPENed with secondary address set to 3 would allow you to change the number of lines per page.

A complex printing program which takes advantage of all the above functions of your printer could have 8 files open at the same time. Remember that there is a limit of 10 files that can be open at any one time. As an aid to avoid initial programming confusion, it is highly recommended that you adapt the following file number conventions in all your printer programs. (All examples in this guide keep to this method).

<u>Logical File Number (lfn)</u>	<u>Secondary Address (sa)</u>	<u>Example</u>
4	0	OPEN 4,4 (OPEN 4,4,0)
3	3	OPEN 3,4,3
5	5	OPEN 5,4,5
6	6	OPEN 6,4,6
7	7	OPEN 7,4,7
10	10	OPEN10,4,10
19	19	OPEN19,4,19
20	20	OPEN20,4,20

The reason that file#4 has been assigned a secondary address of 0 is that a logical file number of 0 is not allowed. Notice that the secondary address was left off the OPEN command for file#4 - the 0 need not be as it is the default value. Since we assigned file#4 to secondary address of 0.

NOTE: A user may have SA=0 and 7 open at the same time to print mixed Business and Graphic Character Mode.

Let's look at each secondary address:-

1. SA=0: Printing in Graphic Mode

This secondary address causes the printer to print ASCII data in Graphic (Graphic/Upper-case characters) Mode. As mentioned before, this secondary address is the default value and need not be coded.

The format for this command is as follows:

```
OPEN 4,4 (OPEN 4,4,0)
PRINT#4 <,variable/data>
CLOSE4
```

Try the following sample programs:

Example 1:

```
100 REM SA=0...PRINTING IN GRAPHIC MODE
110 OPEN4,4 :REM OPEN4,4,0
120 DC$=CHR$(28)
130 PRINT#4,DC$;CHR$(0);:REM BLACK
140 :
150 PRINT#4,"SECONDARY ADDRESS 0"
160 A$="PRINTING IN GRAPHIC MODE"
170 PRINT#4,A$
180 PRINT#4,123456789
190 FOR I=64 TO 95
200 PRINT#4,CHR$(I);
210 NEXT I
220 :
230 FOR J=1 TO 15:PRINT#4:NEXT J
240 CLOSE4
```

RUN

```
SECONDARY ADDRESS 0
PRINTING IN GRAPHIC MODE
123456789
@ABCDEFGHIJKLMN O PQRSTU VWXYZ [ \ ] ^ _
```

The second sample program will print ASCII code table in Graphic Mode. The ASCII \$00-\$1F and \$80-\$9F are special control characters, so the this code table doesn't print these code.

Example 2:

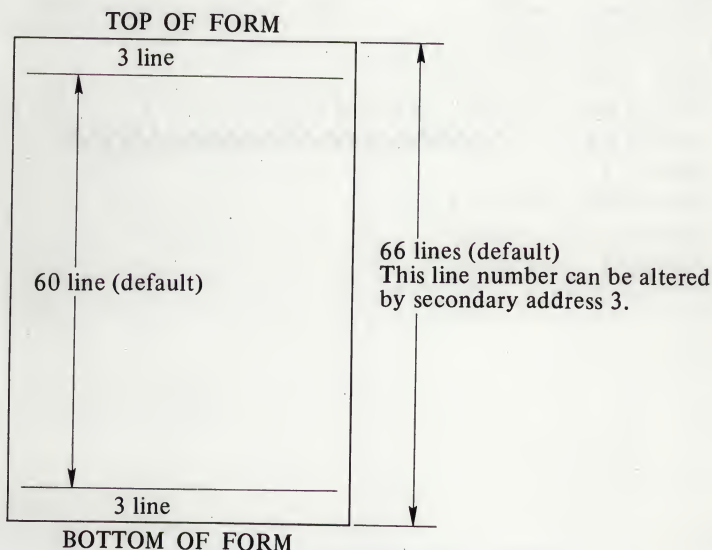
```
100 REM SA=0...PRINTING IN GRAPHIC MODE
110 OPEN 4,4 :REM OPEN 4,4,0
120 DC$=CHR$(20)
130 PRINT#4,DC$;CHR$(2); :REM RED
140 :
150 SP$=" "
160 A$="0123456789ABCDEF"
170 PRINT#4,SP$;SP$;
180 FOR I=0 TO 15
190 PRINT#4,MID$(A$,I+1,1);SP$;
200 NEXT I
210 PRINT#4
220 :
230 FOR J=0 TO 15
240 PRINT#4,MID$(A$,J+1,1);SP$;
250 PRINT#4,SP$;SP$;SP$;SP$;
260 PRINT#4,CHR$(J+32);SP$;CHR$(J+48);SP$;
270 PRINT#4,CHR$(J+64);SP$;CHR$(J+80);SP$;
280 PRINT#4,CHR$(J+96);SP$;CHR$(J+112);SP$;
290 PRINT#4,SP$;SP$;SP$;SP$;
300 PRINT#4,CHR$(J+160);SP$;CHR$(J+176);SP$;
310 PRINT#4,CHR$(J+192);SP$;CHR$(J+208);SP$;
320 PRINT#4,CHR$(J+224);SP$;CHR$(J+240)
330 NEXT J
340 FOR K=1 TO 7:PRINT#4:NEXT K
350 CLOSE4
```

RUN

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0			0	Q	P	-	7				r	-	7		r	
1		!	1	A	Q	•	•				■	+	•	•	■	+
2		"	2	B	R		-				■	+		-	■	+
3		#	3	C	S	-	♥				7	+	-	♥	-	+
4		\$	4	D	T	-					-		-		-	
5		%	5	E	U	-	/						-	/		
6		&	6	F	V	-	X				■		-	X	■	
7		'	7	G	W		0					-		0		-
8		(8	H	X		+				■	-		+	■	-
9)	9	I	Y	~					■	-	~		■	-
A		*	:	J	Z	~	♦]	~	♦]
B		+	;	K	[~	+				+	■	~	+	+	■
C		,	<	L	£	L	■				■	■	L	■	■	■
D		-	=	M	J	\					L]	\		L]
E		.	>	N	†	/	π				7	■	/	π	7	■
F		/	?	0	+	Γ	▼				-	■	Γ	▼	-	π

3. SA=3: Setting the Number of Lines Per Page

This secondary address allows you to vary the number of printed lines per page. In order for this paging option to take effect, you must turn 'PAGING ON' with the special character described on SPECIAL CONTROL CHARACTERS (Section ⑤). When paging is on and the secondary address is not implemented, the default number of lines per page is 66 including three blank lines at the top of the page and three blank lines at the bottom of the page.



The format for this command is as follows:

```
OPEN 3,4,3
PRINT#3 ,CHR$( <the number of line> )
CLOSE3
```

Try the following sample programs:

This program will print 16 lines per page. At first, you must set first line to the printer. When you ready, PRESS any key. The printer is continuous until you will PRESS RUN/STOP key.

Example :

```
100 REM SA=3...SETTING THE NUMBER OF
110 REM LINES PER PAGE
120 OPEN 4,4,0 :REM PRINT IN GRAPHIC MODE
130 DC$=CHR$(20)
140 PRINT#4,DC$;CHR$(3) :REM CYAN
150 OPEN 3,4,3 :REM OPEN SET LINES PER PAGE
160 :
```



```

170 PRINT"SET FIRST LINE"
180 PRINT"PRESS ANY KEY WHEN READY."
190 GET A$:IF A$="" THEN 190
200 PRINT"PRINT START"
210 PRINT"PRESS STOP KEY WHEN YOU SATISFIED"
220 :
230 A=1:L=16
240 SP$=""
250 PRINT#3,CHR$(L) :REM SET LINES PER PAGE
260 PRINT#4,CHR$(147);
270 FOR I=A TO A+39
280 PRINT#4,"LINE";RIGHT$(SP$+STR$(I),4);
290 PRINT#4," "
300 NEXT I
310 PRINT#4,CHR$(19);
320 A=A+40:GOTO260
330 CLOSE4:CLOSE3












```

RUN

```

LINE 1
LINE 2
LINE 3
LINE 4
LINE 5
LINE 6
LINE 7
LINE 8
LINE 9
LINE 10
LINE 11
LINE 12
LINE 13
LINE 14
LINE 15
LINE 16

```


LINE 41 
LINE 42 
LINE 43 
LINE 44 
LINE 45 
LINE 46 
LINE 47 
LINE 48 
LINE 49 
LINE 50 
LINE 51 

Press RUN/STOP key

3. SA=5: Defining a Programmable Character

A secondary address of 5 allows you a design your own character. This programmable character is initialized with this secondary address.

The format for this command is as follows:

```
OPEN 5,4,5
PRINT#5,CHR$(D1)CHR$(D2)CHR$(D3)CHR$(D4)CHR$(D5)
      CHR$(D6)CHR$(D6)CHR$(D7)CHR$(D8)
CLOSE5
OPEN 4,4
PRINT#4,CHR$(254).....Print the Custom Character
CLOSE4
```

Example 1:

Suppose you wanted to program the special symbol in Commodore Business Machines's logo, which is:



Layout a 8*8 matrix this corresponds to the format on the print head. To the left of the matrix, write the binary bit value of each line. Use dots, one per square to create your character. Then add up the binary bits (indicated by your dots) in each column. These totals are used in the DATA statement in your program.

1								
2			●	●				
4		●			●	●		
8	●				●			
16	●							
32	●				●			
64		●			●	●		
128			●	●				
	56	68	130	130	108	68	0	0

←TOTALS

The DATA statement in your program will read:

```
DATA 56,68,130,130,108,68,0,0
```

The program shown in the printout in the next example writes the Commodore logo five times. It creates a string with the CHR\$

value of the column totals and passes the string to the printer with SA=5 .

```
100 rem sa=5...defining a programmable
110 rem character
120 :
130 data 56,68,130,130,108,68,0,0
140 open 5,4,5 :rem define a prg char
150 for i=1 to 8:read a:a$=a$+chr$(a):next i
160 print#5,a$
170 :
180 open 7,4,7 :rem print in business mode
190 dc$=chr$(20) :pc$=chr$(254)
200 en$=chr$(14) :de$=chr$(15)
210 :
220 for j=0 to 7
230 if j=1 or j= 7 then 280
240 print#7,dc$;chr$(j); :rem set color
250 print#7,en$;pc$;
260 print#7," Commodore";
270 print#7,de$
280 next j
290 :
300 for k=1 to 15:print#7:next k
310 close7:close5
```

The following is a description of the program shown above:

Line No.

100-110 PROGRAM TITLE

120 Separates program segments

130 Data used for the custom character

140 OPEN 5,4,5

└ Communicates to the printer what function
 you want executed. A secondary address of 5
 indicates that you want something stored in
 the printer's memory.

└ The device number (the printer's)

└ The logical file number

150 The FOR NEXT loop contained in this line reads the

data contained in line 130 and constructs a string a\$.
a\$ contains the necessary information to print the
Commodore logo.

```
160 Stores a$ in printer memory.
170 Separates program segments
180 Open the print file for printing in Business Character
    Mode.
190 Set dc$ to CHR$(20), pc$ to CHR$(254)
200 Set en$ to CHR$(14), de$ to CHR$(15)
210 Separates program segments
220-280 Prints the Commodore logo and "Commodore" in different
       color.
230 Avoids white and yellow
240 Set color
250 Enhances characters and prints the programmed
    character
260 Prints the string " Commodore"
270 Enhance off characters
290 Separates program segments
300 Fifteen times carriage return
310 Closes the two files opened by the program.
```

After typing RUN, you get this result:

```
Q Commodore
Q Commodore
Q Commodore
Q Commodore
Q Commodore
Q Commodore
```

Multiple programmable characters in the same line can only be made by overprinting. The programmable character cannot be changed when a line wraps around to the next line.

Example 2:

```
100 rem sa=5...defining a programmable
110 rem character
120 :
130 data 56,68,130,130,108,68,0,0
135 data 0,0,68,108,130,130,68,56
```

```

140 open 5,4,5 :rem define a prg char
150 for i=1 to 8:read a:a$=a$+chr$(a):next i
155 for i=1 to 8:read b:b$=b$+chr$(b):next i
170 :
180 open 7,4,7 :rem print in business mode
190 dc$=chr$(20) :pc$=chr$(254) :cs$=chr$(141)
200 en$=chr$(14) :de$=chr$(15)
210 :
220 for j=0 to 7
230 if j=1 or j= 7 then 280
240 print#7,dc$;chr$(j); :rem set color
250 print#5,a$;print#7,en$;pc$;de$;cs$;
260 print#5,b$;print#7,en$;" ";pc$;de$;cs$;
270 print#5,a$;print#7,en$;" ";pc$;de$;cs$;
275 print#5,b$;print#7,en$;" ";pc$;de$;cs$
280 next j
290 :
300 for k=1 to 15:print#7:next k
310 close7:close5

```

run

```

G DG D
G DG D
G DG D
G DG D
G DG D
G DG D

```


4. SA=6: Setting Spacing Between Lines

A secondary address of 6 controls the number of steps between successive lines of print. There are 120 steps per lines. To calculate the declared value, divide 120 by the number of lines per inch required (e.g. a declared value of 40 produces 3 lines per inch, a declared value of 60 produces 2 lines per inch).

The default value is 20, which produced the standard 6 lines per inch.

The declared value is used as a character code (see line 160 in example below - CHR\$(12) - created by dividing 120 by the 10 lines). Therefore when the program (example 1) is run there will be no spaces between the lines of the block.

The format for this command is as follows:

```
OPEN 6,4,6
PRINT#6 <,CHR$(120/lines per inch)>
CLOSE6
```

Try the following sample program.

Example 1:

This program will print box.

```
100 REM SA=6...SETTING SPACING BETWEEN LINES
110 :
120 OPEN 4,4 :REM PRINT IN GRAPHIC MODE
130 DC$=CHR$(20)
140 PRINT#4,DC$;CHR$(6);:REM BLUE
150 OPEN 6,4,6 :REM SET LINE SPACING
160 PRINT#6,CHR$(12):REM CONT'D VERTICAL DOT
170 :
180 PRINT#4," "
190 PRINT#4," | "
200 PRINT#4," | "
210 PRINT#4," | "
220 PRINT#4," | "
230 PRINT#4," | "
240 PRINT#4," | "
250 PRINT#4," | "
260 PRINT#4," | "
270 PRINT#4," | "
280 :
290 PRINT#6,CHR$(20) :REM ORIGINAL SPACING
300 :
310 FOR I=1 TO 15:PRINT#4:NEXT I
```


320 CLOSE4:CLOSE6

RUN

Example 2:

This program will print string "HHHHHHHHH...HHH" in variable line spacing. If I is over 127 in PRINT#6,CHR\$(I) , overprinting is occures.

```
100 REM SA=6...SETTING SPACING BETWEEN LINES
110 :
120 OPEN 4,4 :REM PRINT IN GRAPHIC MODE
130 DC$=CHR$(20)
140 PRINT#4,DC$;CHR$(0);:REM BLACK
150 OPEN 6,4,6:REM SET LINE SPACING
160 FOR I=0 TO 30
170 PRINT#6,CHR$(I);
180 PRINT#4,RIGHT$(" "+STR$(I),3);" ";
190 PRINT#4,"_____ "
200 NEXT I
210 :
220 PRINT#6,CHR$(20):REM ORIGINAL SPACING
230 :
240 FOR J=1 TO 15:PRINT#4,NEXT J
250 CLOSE4:CLOSE6
```

RUN

0	
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	

20 _____
21 _____
22 _____
23 _____
24 _____
25 _____
26 _____
27 _____
28 _____
29 _____
30 _____

5. SA=7: Printing in Business Mode

To change the printer to print on Business (Upper/Lower-Case Character) Mode, you must open the printer file on SA=7:

The format for this command is as follows:

```
OPEN 4,4,7
PRINT#4 <,variable/data>
CLOSE4
or
OPEN 7,4,7
PRINT#7 <,variable/data>
CLOSE7
```

Try the following sample programs:

Example 1:

```
100 rem sa=7...printing in business mode
110 open7,4,7
120 dc$=chr$(20)
130 print#7,dc$;chr$(2);:rem red
140 :
150 print#7,"secondary address 7"
160 a$="printing in business mode"
170 print#7,a$
180 print#7,123456789
190 for i=64 to 95
200 print#7,chr$(i);
210 next i
220 :
230 for j=1 to 15:print#7:next j
240 close7
```

run

```
secondary address 7
printing in business mode
123456789
@abcdefghijklmnopqrstuvwxyz[F]↑←
```

The second sample program will print ASCII code table in Graphic Mode. The ASCII \$00-\$1F and \$80-\$9F are special control characters, so the code table doesn't print these code.

Example 2:

```

100 rem sa=7...printing in business mode
110 open 7,4,7
120 dc$=chr$(20)
130 print#7,dc$;chr$(3);:rem cyan
140 sp$=" "
150 a$="0123456789ABCDEF"
160 print#7,sp$;sp$;
170 for i=0 to 15
180 print#7,mid$(a$,i+1,1);sp$;
190 next i
200 print#7
210 :
220 for j=0 to 15
230 print#7,mid$(a$,j+1,1);sp$;
240 print#7,sp$;sp$;sp$;sp$;
250 print#7,chr$(j+32);sp$;chr$(j+48);sp$;
260 print#7,chr$(j+64);sp$;chr$(j+80);sp$;
270 print#7,chr$(j+96);sp$;chr$(j+112);sp$;
280 print#7,sp$;sp$;sp$;sp$;
290 print#7,chr$(j+160);sp$;chr$(j+176);sp$;
300 print#7,chr$(j+192);sp$;chr$(j+208);sp$;
310 print#7,chr$(j+224);sp$;chr$(j+240)
320 next j
330 for k=1 to 7:print#7:next k
340 close4

```

run

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0		0	@	p	-	P						r	-	P		r
1		!	1	a	q	A	Q				┌	└	A	Q		└
2		"	2	b	r	B	R				┌	└	B	R		└
3		#	3	c	s	C	S				┌	└	C	S		└
4		\$	4	d	t	D	T				┌	└	D	T		└
5		%	5	e	u	E	U				┌	└	E	U		└
6		&	6	f	v	F	V				┌	└	F	V		└
7		'	7	g	w	G	W				┌	└	G	W		└
8		(8	h	x	H	X				┌	└	H	X		└
9)	9	i	y	I	Y				┌	└	I	Y		└
A		*	:	j	z	J	Z				┌	└	J	Z		└
B		+	;	k	[K	+				┌	└	K	+		└
C		,	<	l	£	L	£				┌	└	L	£		└
D		-	=	m]	M]				┌	└	M]		└
E		.	>	n	†	N	†				┌	└	N	†		└
F		/	?	o	+	O	+				┌	└	O	+		└

6. SA=10: Resetting the Printer

To reset the printer, you will send a secondary address of 10.

The format for this command is as follows:

```
OPEN 10,4,10
PRINT#10
CLOSE10
```

Example 1:

```
100 REM SA=10...RESETTING THE PRINTER
110 OPEN 10,4,10
120 PRINT#10
130 CLOSE10
```

RUN

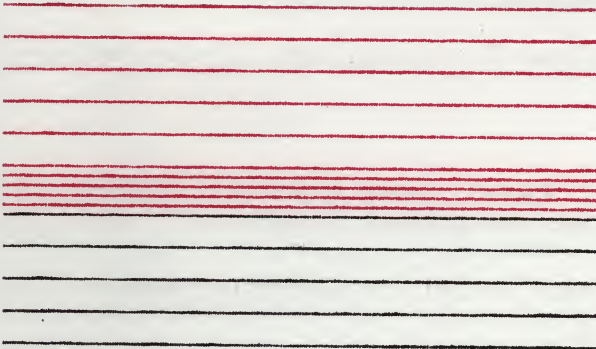
The print head will move about three inches to the right, then return back to the print home position.

Example 2:

```
100 REM SA=10...RESETTING THE PRINTER
110 :
120 OPEN10,4,10:REM RESET THE PRINTER
130 OPEN 4,4,0 :REM PRINT IN GRAPHIC MODE
140 OPEN 6,4,6 :REM SET LINE SPACING
150 :
160 DC$=CHR$(20)
170 PRINT#4,DC$;CHR$(4);:REM PURPLE
180 :
190 GOSUB300
200 :
210 PRINT#6,CHR$(6)
220 GOSUB300
230 :
240 PRINT#10
250 GOSUB300
260 :
270 FOR I=1 TO 15:PRINT#4:NEXT I
280 CLOSE4:CLOSE6:CLOSE10
290 END
300 :
310 FOR I=1 TO 5
320 PRINT#4,"_____"
```


330 NEXT I
340 RETURN

RUN



7. SA=19: Enable 2-pass Mode In Scan Mode Printing

A secondary address of 19 enables 2-pass mode printing in the scan mode bit image printing. The 2-pass mode is especially effective when printing all dots in a given color over all of a given line because the amount of ink transferred from an inker to ribbon is well suited to that sort of printing.

It is highly recommended to use this mode in high-resolution graphics.

Example 1:

```
100 REM SA=19...PRINT SCAN MODE BIT IMAGE
110 REM PRINTING IN 2-PASS MODE
120 :
130 OPEN 4,4 :REM PRINT IN GRAPHIC MODE
140 ESC$=CHR$(27)
150 OPEN 19,4,19
160 PRINT#19 :REM 2-PASS MODE
170 CLOSE19
180 :
185 PRINT"□"
190 PRINT#4,ESC$;"C";"256";"064";
200 FOR K=0 TO 7
210 FOR J=0 TO 7
220 FOR I=0 TO 255
230 A$=CHR$(K)
240 PRINT#4,A$;
245 PRINT"█"K,J,I
250 NEXT I
260 NEXT J
270 NEXT K
280 :
290 FOR L=1 TO 15:PRINT#4:NEXT L
300 CLOSE4
```

RUN



8. SA=20: Disable 2-pass Mode In Scan Mode Printing

A secondary address of 20 disables 2-pass mode printing in the scan mode bit image printing.

Example 1:

```
100 REM SA=20...CANCEL 2-PASS MODE
120 :
130 OPEN 4,4 :REM PRINT IN GRAPHIC MODE
140 ESC$=CHR$(27)
150 OPEN 20,4,20
160 PRINT#20 :REM CANCEL 2-PASS MODE
170 CLOSE20
180 :
185 PRINT"□"
190 PRINT#4,ESC$;"C";"256";"064";
200 FOR K=0 TO 7
210 FOR J=0 TO 7
220 FOR I=0 TO 255
230 A$=CHR$(K)
240 PRINT#4,A$;
245 PRINT"█"K,J,I
250 NEXT I
260 NEXT J
270 NEXT K
280 :
290 FOR L=1 TO 15:PRINT#4:NEXT L
300 CLOSE4
```

RUN



F. SPECIAL CONTROL CHARACTER FUNCTIONS

Special control characters can be used to change the mode of printing (Table 1 contains a summary of the special control characters). Printer control characters are inserted in the data stream transmitted to the printer via secondary address 0, 7.

TABLE 1

Special Control Character Summary

Printer function	CHR\$ Code	Standard ASCII	Keyboard entry
Enhance ON	CHR\$(14)	SOH	NA
Enhance OFF	CHR\$(15)		NA
Paging ON	CHR\$(147)		SHIFT & CLR/HOME
Paging OFF	CHR\$(19)	DC3	CLR/HOME
Reverse ON	CHR\$(18)	DC2	OFF/RVS
Reverse OFF	CHR\$(146)		SHIFT & OFF/RVS
Carriage Return	CHR\$(13)	CR	RETURN
Line Feed	CHR\$(10)	LF	NA
Carriage Return with no line feed	CHR\$(141)		NA
Print in Business Mode	CHR\$(17)	DC1	Cursor Down
Print in Graphic Mode	CHR\$(145)		Cursor Up
Quote	CHR\$(34)	"	Quote
Form Feed	CHR\$(12)	FF	NA
Tab Setting the Print Head	CHR\$(16);CHR\$(NH)CHR\$(NL)		
Repeat 8-bit Graphic Selected	CHR\$(28);CHR\$(GRAPHIC DATA)		
Select Color	CHR\$(20);CHR\$(COLOR DATA)		
Horizontal Scan	CHR\$(27);"C";"NNN";"MMM";CHR\$(GRAPHIC DATA) ..		
8-bit graphic print	CHR\$(27);CHR\$(16);CHR\$(NH)CHR\$(NL)		
Specify Dot Address (must follow POS code)			

1. Enhance ON/OFF CHR\$(14)/CHR\$(15)

Your color printer normally generates a character using dot matrix that is 8 dots high and 8 dots wide. If you send an ASCII CHR\$(14) character within a data output PRINT# statements parameter, all characters following the CHR\$(14) are printed double-width using dot matrix that is 8 dots high and 16 dots wide.

An ASCII CHR\$(15) character cancels the character enhancement specified by preceding CHR\$(14) character.

The example below shows Enhance ON/OFF functions.

```
100 REM CHR$(14)...ENHANCE ON
110 REM CHR$(15)...ENHANCE OFF
120 OPEN 4,4 :REM PRINT IN GRAPHIC MODE
130 EN$=CHR$(14):DE$=CHR$(15)
140 DC$=CHR$(20)
150 A$=" COMMODORE "
160 :
170 FOR I=0 TO 7
180 IF I=1 OR I=7 THEN 210
190 PRINT#4,DC$;CHR$(I);
200 PRINT#4,EN$;A$;DE$;A$
210 NEXT I
220 :
230 FOR J=1 TO 15:PRINT#4:NEXT J
240 CLOSE4
```

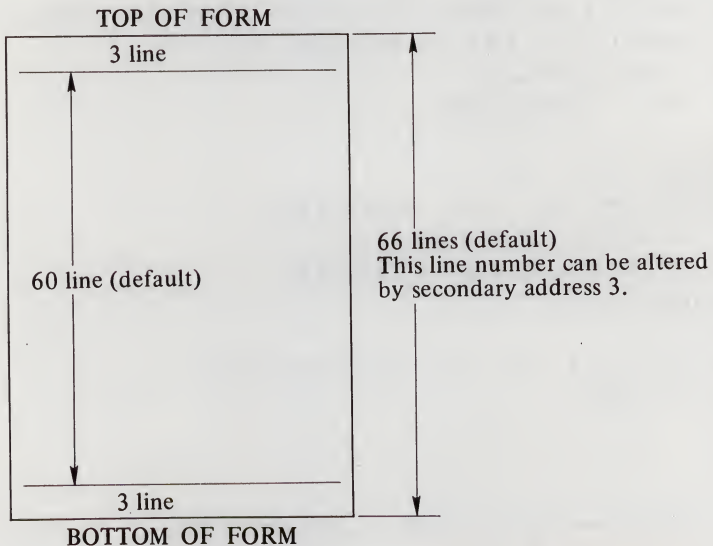
RUN

COMMODORE	COMMODORE
COMMODORE	COMMODORE
COMMODORE	COMMODORE
COMMODORE	COMMODORE
COMMODORE	COMMODORE
COMMODORE	COMMODORE

2. Paging ON/OFF CHRS(147)/CHRS(19)

Paging can be turened on with your program by means of Paging ON Code [CHRS(147)]. Otherwise, printing is continuous. The paging function provides for 66 lines per page including three blank lines at the top of the page and three blank lines at the bottom. The number of the lines per page may be altered by using the secondary address of 3 option (See SA=3) . When paging is in effect, the Paging OFF Code [CHRS(19)] performs paging off and a top of next form. (See the example program of SA=3).

NOTE: Paging ON code [CHRS(147)] resets internal lines counter to zero each time it is sent. You may set 'top of form' start position at any time via this command.



When you want to list up a long line program, you can get paging functioned program list by using Paging ON code (try on the General Test Program in APPENDIX C). Key in the program into your computer, and list on the following commands.

```
OPEN 4,4:CMD4,CHRS(147):LIST
```


3. Selecting Character mode in local CHR\$(17)/CHR\$(145)

By sending the Cursor Down code [CHR\$(17)] to your printer, the characters will be printed in Business Mode until either a carriage return or Cursor Up Code [CHR\$(145)] is detected.

By sending the Cursor Up code [CHR\$(145)] to your printer, the characters will be printed in Graphic Mode until either a carriage return or Cursor Down Code [CHR\$(17)] is detected.

Example 1:

```
100 REM CHR$(17) ...LOCAL BUSINESS MODE
110 REM CHR$(145)...LOCAL GRAPHIC MODE
120 CD$=CHR$(17):CU$=CHR$(145)
130 DC$=CHR$(20)
140 :
150 OPEN 7,4,7 :REM PRINT IN BUSINESS MODE
160 PRINT#7,DC$;CHR$(0);CU$;"♠      ";
170 PRINT#7,DC$;CHR$(3);CD$;"SPADE"
180 PRINT#7,DC$;CHR$(4);CU$;"♥      ";
190 PRINT#7,DC$;CHR$(3);CD$;"HEART"
200 PRINT#7,DC$;CHR$(4);CU$;"♦      ";
210 PRINT#7,DC$;CHR$(3);CD$;"DIAMOND"
220 PRINT#7,DC$;CHR$(0);CU$;"♣      ";
230 PRINT#7,DC$;CHR$(3);CD$;"CLUB"
240 PRINT#7:PRINT#7
250 CLOSE7
260 :
270 OPEN 4,4,0 :REM PRINT IN GRAPHIC MODE
280 PRINT#4,DC$;CHR$(3);CD$;"SPADE  ";
290 PRINT#4,DC$;CHR$(0);CU$;"♠"
300 PRINT#4,DC$;CHR$(3);CD$;"HEART  ";
310 PRINT#4,DC$;CHR$(4);CU$;"♥"
320 PRINT#4,DC$;CHR$(3);CD$;"DIAMOND ";
330 PRINT#4,DC$;CHR$(4);CU$;"♦"
340 PRINT#4,DC$;CHR$(3);CD$;"CLUB   ";
350 PRINT#4,DC$;CHR$(0);CU$;"♣"
360 :
370 FOR I=1 TO 15:PRINT#4:NEXT I
380 CLOSE4
```

RUN

♠	spade
♥	heart
♦	diamond
♣	club

spade	♠
heart	♥
diamond	♦
club	♣

Example 2:

```

100 rem "♠":chr$(17) ...local business mode
110 rem "♥":chr$(145)...local graphic mode
120 open 7,4,7 :rem print in business mode
130 dc$=chr$(20)
140 print#7,dc$;chr$(2);:rem red
150 :
160 print#7,"LOGO M ♠ | ♠8/8 vertical"
170 print#7,"SHIFT Y ♥Y ♥7/8 vertical"
180 print#7,"SHIFT H ♥H ♥6/8 vertical"
190 print#7,"SHIFT I ♠I ♠5/8 vertical"
200 print#7,"SHIFT B ♥B ♥4/8 vertical"
210 print#7,"SHIFT G ♥G ♥3/8 vertical"
220 print#7,"SHIFT T ♥T ♥2/8 vertical"
230 print#7,"LOGO G ♠ ♠1/8 vertical"
240 :
250 for i=1 to 15:print#7:next i
260 close7

```

run

```

LOGO M | 8/8 vertical
SHIFT Y | 7/8 vertical
SHIFT H | 6/8 vertical
SHIFT I | 5/8 vertical
SHIFT B | 4/8 vertical
SHIFT G | 3/8 vertical
SHIFT T | 2/8 vertical
LOGO G | 1/8 vertical

```

4. Reverse ON/OFF CHR\$(18)/CHR\$(146)

By sending the code [CHR\$(18)] to your printer, you have turned on the REVERSE FIELD mode. This prints white letters on a colored background.

By sending the code [CHR\$(146)] to your printer turns off the REVERSE FIELD mode.

Example 1:

```
100 REM CHR$(18) ...REVERSE ON
110 REM CHR$(146)...REVERSE OFF
120 RV$=CHR$(18):RO$=CHR$(146)
130 DC$=CHR$(20)
140 A$=" REVERSE ON ":B$=" REVERSE OFF "
150 :
160 OPEN 4,4 :REM PRINT IN GRAPHIC MODE
170 FOR I=0 TO 7
180 IF I=1 OR I=7 THEN 210
190 PRINT#4,DC$;CHR$(I);
200 PRINT#4,RV$;A$;RO$;B$;RV$;A$
210 NEXT I
220 :
230 FOR I=1 TO 15:PRINT#4:NEXT I
240 CLOSE4
250 OPEN10,4,10:PRINT#10:CLOSE10:REM RESET
```

RUN

REVERSE ON	REVERSE OFF	REVERSE ON
REVERSE ON	REVERSE OFF	REVERSE ON
REVERSE ON	REVERSE OFF	REVERSE ON
REVERSE ON	REVERSE OFF	REVERSE ON
REVERSE ON	REVERSE OFF	REVERSE ON
REVERSE ON	REVERSE OFF	REVERSE ON

Example 2:

```
100 REM "▣" : CHR$(18) ...REVERSE ON
110 REM "▣" : CHR$(146)...REVERSE OFF
120 OPEN 4,4:REM PRINT IN GRAPHIC MODE
130 DC$=CHR$(20)
140 :
150 FOR I=0 TO 7
160 IF I=1 OR I=7 THEN 210
170 PRINT#4,DC$;CHR$(I);
180 PRINT#4,"▣ REVERSE ON "; "▣ REVERSE OFF ";
```

```

190 PRINT#4,"REVERSE ON ","REVERSE OFF ";
200 PRINT#4
210 NEXT I
220 :
230 FOR J=1 TO 15:PRINT#4:NEXT J
240 CLOSE4
250 OPEN16,4,10:PRINT#10:CLOSE10:REM RESET

```

REVERSE ON	REVERSE OFF	REVERSE ON	REVERSE OF
REVERSE ON	REVERSE OFF	REVERSE ON	REVERSE OF
REVERSE ON	REVERSE OFF	REVERSE ON	REVERSE OF
REVERSE ON	REVERSE OFF	REVERSE ON	REVERSE OF
REVERSE ON	REVERSE OFF	REVERSE ON	REVERSE OF
REVERSE ON	REVERSE OFF	REVERSE ON	REVERSE OF

5. Line Feed/Carriage Return/Carriage Stand

CHR\$(10)/CHR\$(13)/CHR\$(141)

By sending LF Code [CHR\$(10)] to your printer, all data in the print buffer is printed and the paper is advanced one line.

By sending CR Code [CHR\$(13)] to your printer, all data in the print buffer is printed and the paper is advanced one line.

By sending CS Code [CHR\$(141)] to your printer, a Carriage Return without Line Feed is executed. This allows overprinting on the same line.

A Carriage Return turns off REVERSE FIELD and quote mode.

Example :

```
100 REM CHR$(10) ...LINE FEED
110 REM           WITH CARRIAGE RETURN
120 REM CHR$(13) ...CARRIAGE RETURN
130 REM           WITH LINE FEED
140 REM CHR$(141)...CARRIAGE RETURN
150 REM           WITHOUT LINE FEED
160 LF$=CHR$(10):CR$=CHR$(13):CS$=CHR$(141)
170 DC$=CHR$(20)
180 OPEN 4,4 :REM PRINT IN GRAPHIC MODE
190 DIM C(4):C(1)=3:C(2)=4:C(3)=5:C(4)=6
200 :
210 FOR I=1 TO 3
220 PRINT#4,DC$;CHR$(C(I));
230 PRINT#4,"CHR$(10) ...LINE FEED ";
240 PRINT#4,"(WITH CARRIAGE RETURN)";LF$;
250 NEXT I
260 PRINT#4,DC$;CHR$(C(4));
270 FOR J=1 TO 43:PRINT#4,"_";:NEXT J
280 PRINT#4
290 :
300 FOR I=1 TO 3
310 PRINT#4,DC$;CHR$(C(I));
320 PRINT#4,"CHR$(13) ...CARRIAGE RETURN";
330 PRINT#4,"(WITH LINE FEED)";CR$;
340 NEXT I
350 PRINT#4,DC$;CHR$(C(4));
360 FOR J=1 TO 43:PRINT#4,"_";:NEXT J
370 PRINT#4
380 :
390 FOR I=1 TO 3
400 PRINT#4,DC$;CHR$(C(I));
410 PRINT#4,"CHR$(141) ...CARRIAGE RETURN";
420 PRINT#4,"WITHOUT LINE FEED)";CS$;
430 NEXT I
```

```

440 PRINT#4,DC$:CHR$(C(4));
450 FOR J=1 TO 43:PRINT#4,"_":NEXT J
460 PRINT#4
470 :
480 FOR K=1 TO 15:PRINT#4:NEXT K
490 CLOSE4
500 OPEN10,4,10:PRINT#10:CLOSE10:REM RESET

```

RUN

```

CHR$(10) ...LINE FEED (WITH CARRIAGE RETURN)
CHR$(10) ...LINE FEED (WITH CARRIAGE RETURN)
CHR$(10) ...LINE FEED (WITH CARRIAGE RETURN)

```

```

CHR$(13) ...CARRIAGE RETURN(WITH LINE FEED)
CHR$(13) ...CARRIAGE RETURN(WITH LINE FEED)
CHR$(13) ...CARRIAGE RETURN(WITH LINE FEED)

```

```

CHR$(141) ...CARRIAGE RETURNWITHOUT LINE FEED)

```

AUTOMATIC CARRIAGE RETURN

If you attempt to print more than 80 characters (standard size) on a line, a Carriage Return with Line Feed will be forced and overflow characters will be printed on the next line.

Example :

```

100 REM AUTO CARRIAGE RETURN
110 OPEN4,4 :REM PRINT IN GRAPHIC MODE
120 DC$=CHR$(20)
130 :
140 A$="1234567890":B$=""
150 PRINT#4,DC$;CHR$(0);:REM BLACK
160 FOR I=1 TO 8
170 PRINT#4,A$;
180 NEXT I
190 PRINT#4
200 :
210 C=2
220 FOR J=1 TO 12
230 PRINT#4,DC$;CHR$(C);
240 B$=B$+A$
250 PRINT#4,B$
260 C=C+1
262 IF C=1 THEN C=2
265 IF C=7 THEN C=0

```

```
270 NEXT J
280 :
290 FOR K=1 TO 15:PRINT#4:NEXT K
300 CLOSE4
310 OPEN10,4,10:PRINT#10:CLOSE10:REM RESET
```


6. Quotation Marks " or CHR\$(34)

In order to print quotation marks around particular characters, use the character code [CHR\$(34)]. When incorporating quotation marks in your program for print statement purposes, use them logically and effectively, do not place any together unless the program design specifically requires it.

The printed output of the example 1 below will be:

"COMMODORE"

"BUSINESSQUOTE\$

" 0 "

Line 160 forms the print routine for "COMMODORE"

Line 180 will print "BUSINESSQUOTE\$" because both BUSINESS and CHR\$(34) are enclosed within their own quotes.

Line 200 will print " 0 " because the word MACHINES is taken as a numeric variable since it is not enclosed within quotes and it's contents are naturally zero.

Example 1:

```
100 REM PROGRAM TO ILLUSTRATE THE USE
110 REM OF QUOTE MARKS
120 :
130 OPEN 4,4:REM PRINT IN GRAPHIC MODE
140 DC$=CHR$(20):QUOTE$=CHR$(34)
150 PRINT#4,DC$;CHR$(2) :REM RED
160 PRINT#4,QUOTE$;"COMMODORE";QUOTE$
170 PRINT#4
180 PRINT#4,QUOTE$;"BUSINESSQUOTE$
190 PRINT#4
200 PRINT#4,QUOTE$;MACHINES;QUOTE$
210 PRINT#4
220 FOR I=1 TO 15:PRINT#4:NEXT I
230 CLOSE4
```

If an odd number of quotation marks have been transmitted, control character are made visible. This can be particularly useful when you are making a listing a BASIC program containing control characters in quotation marks.

Example 2: (Graphic Mode)

```
100 REM CONTROL CHARACTERS
110 PRINT"␣ HOME"
120 PRINT"␣ CLR"
130 PRINT"␣ CRSR DOWN"
140 PRINT"␣ CRSR UP"
150 PRINT"␣ CRSR RIGHT"
160 PRINT"␣ CRSR LEFT"
170 PRINT"␣ RVS ON"
180 PRINT"␣ RVS OFF"
190 PRINT"␣ F1"
200 PRINT"␣ F3"
210 PRINT"␣ F5"
220 PRINT"␣ F7"
230 PRINT"␣ F2"
240 PRINT"␣ F4"
250 PRINT"␣ F6"
260 PRINT"␣ F8"
270 PRINT"␣ BLACK"
280 PRINT"␣ WHITE"
290 PRINT"␣ RED"
300 PRINT"␣ CYAN"
310 PRINT"␣ PURPLE"
320 PRINT"␣ GREEN"
330 PRINT"␣ BLUE"
340 PRINT"␣ YELLOW"
350 REM COMMODORE 64
360 PRINT"␣ ORANGE"
370 PRINT"␣ BLOWN"
380 PRINT"␣ LIGHT RED"
390 PRINT"␣ DARK GREY"
400 PRINT"␣ MED GREY"
410 PRINT"␣ LIGHT GREEN"
420 PRINT"␣ LIGHT BLUE"
430 PRINT"␣ LIGHT GREY"
440 OPEN4,4:CMD4:LIST
```

READY.

Example 3: (Business Mode)

```
100 rem control characters
110 print"␣ home"
120 print"␣ clr"
130 print"␣ crsr down"
140 print"␣ crsr up"
150 print"␣ crsr right"
160 print"␣ crsr left"
170 print"␣ rvs on"
180 print"␣ rvs off"
190 print"␣ f1"
200 print"␣ f3"
210 print"␣ f5"
220 print"␣ f7"
230 print"␣ f2"
240 print"␣ f4"
250 print"␣ f6"
260 print"␣ f8"
270 print"␣ black"
280 print"␣ white"
290 print"␣ red"
300 print"␣ cyan"
310 print"␣ purple"
320 print"␣ green"
330 print"␣ blue"
340 print"␣ yellow"
350 rem commodore 64
360 print"␣ orange"
370 print"␣ blown"
380 print"␣ light red"
390 print"␣ dark grey"
400 print"␣ med grey"
410 print"␣ light green"
420 print"␣ light blue"
430 print"␣ light grey"
440 open 7,4,7:cmd7:list
```

ready.

7. Print Position Determination

With the POS Code [CHR\$(16)] you can determine the print start position. This is done by assigning a 2-digit number following the CHR\$(16) (see the examples below).

Example 1:

```
100 REM CHR$(16)...SET PRINT POSITION
110 :
120 OPEN 4,4,0 :REM PRINT IN GRAPHIC MODE
130 DC$=CHR$(20)
140 PO$=CHR$(16)
150 :
160 PRINT#4,DC$;CHR$(0);
170 FOR I=1 TO 4
180 PRINT#4,"0123456789";
190 NEXT I
200 PRINT#4
210 PRINT#4,DC$;CHR$(3);
220 PRINT#4,PO$;CHR$(48)CHR$(56);"COMMODORE";
230 PRINT#4,PO$;CHR$(51)CHR$(48);"PRINTER";
240 :
250 FOR J=1 TO 15:PRINT#4:NEXT J
260 CLOSE4
270 OPEN10,4,10:PRINT#10:CLOSE10:REM RESET
```

RUN

```
0123456789012345678901234567890123456789
      COMMODORE                PRINTER
```

Exanoples 2:

```
100 REM CHR$(16)...SET PRINT POSITION
110 :
120 OPEN 4,4,0 :REM PRINT IN GRAPHIC MODE
130 DC$=CHR$(20)
140 PO$=CHR$(16)
150 :
160 PRINT#4,DC$;CHR$(0);
170 FOR I=1 TO 4
180 PRINT#4,"0123456789";
190 NEXT I
200 PRINT#4
210 PRINT#4,DC$;CHR$(4);
220 PRINT#4,PO$;"08";"COMMODORE";
```



```

230 PRINT#4,PO$;"30";"PRINTER";
240 :
250 FOR J=1 TO 15:PRINT#4:NEXT J
260 CLOSE4
270 OPEN10,4,10:PRINT#10:CLOSE10:REM RESET

```

RUN

```

0123456789012345678901234567890123456789
      COMMODORE          PRINTER

```

The two numbers following the CHR\$(16) code are the print start position of your standard characters. This can be proven using the following example.

Example 3:

```

100 REM CHR$(16)...SET PRINT POSITION
110 :
120 OPEN 4,4,0 :REM PRINT IN GRAPHIC MODE
130 DC$=CHR$(20)
140 PO$=CHR$(16):EN$=CHR$(14):DE$=CHR$(15)
150 :
160 PRINT#4,DC$;CHR$(0);
170 FOR I=1 TO 5
180 PRINT#4,"0123456789";
190 NEXT I
200 PRINT#4
210 PRINT#4,DC$;CHR$(4);
215 PRINT#4,EN$;
220 PRINT#4,PO$;"08";"COMMODORE";
230 PRINT#4,PO$;"30";"PRINTER";
235 PRINT#4,DE$
240 :
250 FOR J=1 TO 15:PRINT#4:NEXT J
260 CLOSE4
270 OPEN10,4,10:PRINT#10:CLOSE10:REM RESET

```

RUN

```

01234567890123456789012345678901234567890123456
      COMMODORE          PRINTER

```

8. Form Feed CHR\$(12)

By sending a Form Feed Code [CHR\$(12)], your printer executes the printing of all the data stored in the print buffer and advances the paper to the next predetermined top of form position.

Example :

```
100 REM CHR$(12)...FORM FEED
110 :
120 PRINT"RESET PRINTER"
130 OPEN 10,4,10
140 PRINT#10      :REM RESET PRINTER
150 CLOSE10
160 :
170 PRINT"RESET FIRST LINE"
180 PRINT"PRESS ANY KEY WHEN READY."
190 GET A$:IF A$="" THEN 190
200 :
210 FF$=CHR$(12):SP$=""
220 DC$=CHR$(20):C=0
230 OPEN 4,4 :REM PRINT IN GRAPHIC MODE
240 INPUT "PRINT LINE NUMBER";L
250 FOR I=1 TO L
260 PRINT#4,DC$;CHR$(C);
270 PRINT#4,"LINE";RIGHT$(SP$+STR$(I),4)
280 C=C+1
290 IF C=1 THEN C=2
300 IF C=7 THEN C=0
310 NEXT I
320 PRINT#4,FF$;
330 CLOSE4
340 OPEN10,4,10:PRINT#10:CLOSE10:REM RESET
```

RUN

Enter this program correctly, and RUN it. According to the display messages, you must set the first line to (for example) paper cutting line, and input the quantity of lines you require printing.

LINE	1
LINE	2
LINE	3
LINE	4
LINE	5
LINE	6
LINE	7
LINE	8
LINE	9
LINE	10
LINE	11

66 lines (default)
This line number can be
altered by SA3.

9. Repeat Graphic Printing

FS;CHR\$(n);CHR\$(Graphic Data)

This codes sequence specifies the repeated printing of graphic data. "n" is a binary number (0 through 255) which specifies the desired number of the printed repetition; followed by one-byte graphic data to be printing repeatedly.

When 0 is specified for "n", it is read as 256. In order to repeat more than 256 times the operator needs to use this code twice more.

Example :

```
100 REM REPEAT GRAPHIC PRINTING
110 REM FS;CHR$(N);CHR$(GRAPHIC DATA)
120 OPEN 4,4 :REM PRINT IN GRAPHIC MODE
130 DC#=CHR$(20):C=0
140 FS#=CHR$(28)
150 :
160 FOR I=0 TO 7
170 PRINT#4,DC#;CHR$(C);:REM SET COLOR
180 P=20+20*I*2:IF P>256 THEN L0=P-255:GOTO210
190 PRINT#4,FS#;CHR$(P);CHR$(243):GOTO230
200 PRINT#4,FS#;CHR$(L0);CHR$(243):GOTO230
210 PRINT#4,FS#;CHR$(255);CHR$(243);
220 PRINT#4,FS#;CHR$(L0);CHR$(243)
230 C=C+1
240 IF C=1 THEN C=2
250 IF C=7 THEN C=0
260 NEXT I
270 :
280 FOR J=1 TO 15:PRINT#4:NEXT J
290 CLOSE4
300 OPEN10,4,10:PRINT#10:CLOSE10:REM RESET
```

RUN



10. Dot Address Determination

ESC POS nH nL

This code specifies print start position in dot units. nH nL is a 2-byte binary number (0 through 639) which indicates the dot where printing starts. When a number greater than 639 is specified, the data is printed from the beginning of the next line.

D8	D7	65	D5	D4	D3	D2	D1	
-----	-----	-----	-----	-----	-----	-----	-----	
0	0	0	0	0	0	P10	P9	higher 2 bits
-----	-----	-----	-----	-----	-----	-----	-----	
P8	P7	P6	P5	P4	P3	P2	P1	lower 8 bits
-----	-----	-----	-----	-----	-----	-----	-----	

Desired number of print start positions can be specified. If a printed start position is specified at a dot where other data is to be printed, overstrike takes place.

Example 1:

```
100 REM DOT POSITION SET
110 OPEN 4,4 :REM PRINT IN GRAPHIC MODE
120 DC$=CHR$(20)
130 PO$=CHR$(16):ESC$=CHR$(27)
140 :
150 FOR I=1 TO 16
160 YO=INT(RND(1)*320):YO$=STR$(YO)
170 YH=INT(YO/256):YL=YO-YH*256
180 C=INT(RND(1)*7)
190 IF C=1 THEN 180
200 IF C=1 THEN 180
210 PRINT#4,DC$;CHR$(C); :REM SET COLOR
220 PRINT#4,ESC$;PO$;CHR$(YH)CHR$(YL);"*";
230 PRINT#4,YO$
240 NEXT I
250 :
260 FOR J=1 TO 15:PRINT#4:NEXT J
270 CLOSE 4
280 OPEN 10,4,10:PRINT#10:CLOSE 10:REM RESET
```

RUN

```

* 155
* 56
* 233
* 199
* 146
* 170
* 24
* 86
* 49
* 187
* 30
* 268
* 85
* 130
* 45
* 218

```

Example 2:

```

100 REM SIN CURVE
110 OPEN4,4 :REM PRINT IN GRAPHIC MODE
120 DC$=CHR$(20)
130 PO$=CHR$(16):ESC$=CHR$(27)
140 EN$=CHR$(14):DE$=CHR$(15)
150 :
160 OPEN6,4,6
170 PRINT#6,CHR$(12):REM CONTINUOUS PRINT
180 :
190 CN=23:AM=16:OF=4
200 A$="-":FOR I=0 TO CN+AM:A$=A$+"-":NEXT I
210 SP$=""
220 PRINT#4,DC$;CHR$(4);:REM PURPLE
230 PRINT#4,EN$;" SIN CURVE";DE$
240 PRINT#4
250 PRINT#4,DC$;CHR$(8);:REM BLACK
260 PRINT#4,LEFT$(SP$,OF-1);"X";
270 PRINT#4,LEFT$(SP$,CN-AM-OF-1);"-1";
280 PRINT#4,LEFT$(SP$,AM-1);"0";
290 PRINT#4,LEFT$(SP$,AM-1);"1"
300 PRINT#4,A$
310 :
320 FOR J=0 TO 360 STEP 10
330 I$=RIGHT$(SP$+STR$(J),OF)

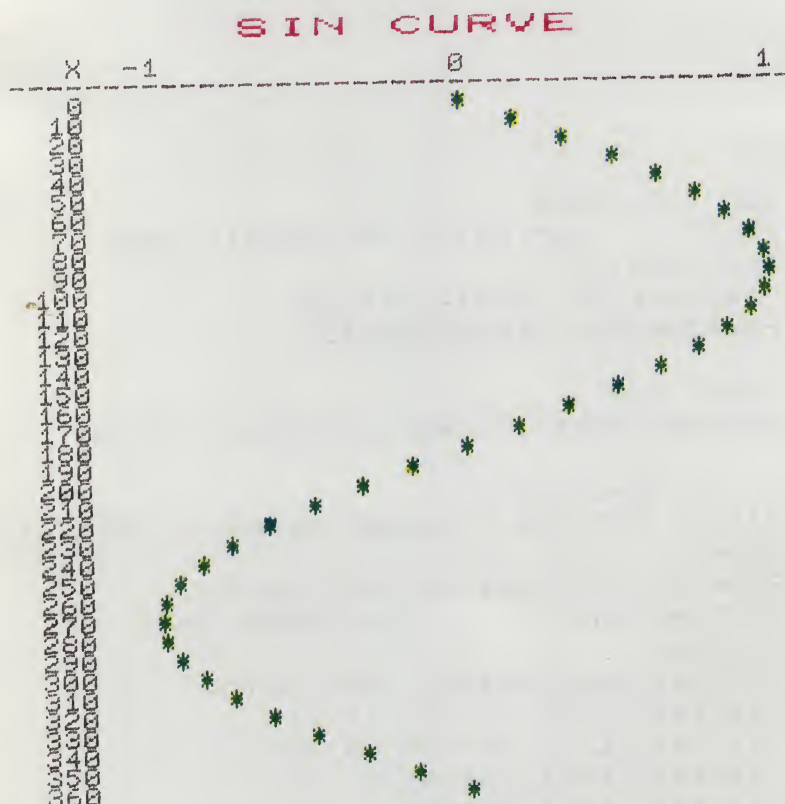
```

```

340 Y0=CN*8+AM*8*SIN(J*π/180)
350 YH=INT(Y0/256):YL=Y0-YH*256
360 PRINT#4,DC$;CHR$(0);:REM BLACK
370 PRINT#4,I$;:REM SCALE
380 PRINT#4,DC$;CHR$(5);:REM GREEN
390 PRINT#4,ESC$;PO$;CHR$(YH)CHR$(YL);"*"
400 NEXT J
410 :
420 PRINT#6,CHR$(20):REM RESET TO ORIG. SPACING
430 :
440 FOR K=1 TO 15:PRINT#4:NEXT K
450 CLOSE4
460 OPEN10,4,10:PRINT#10:CLOSE10:REM RESET

```

RUN



APPENDICES

APPENDIX A: MCS-801 COLOR PRINTER SPECIFICATIONS

1. Printing Specifications

a. Print Method	Impact type dot matrix (4-hammer method)
b. Printing direction	Unidirectional (left to right)
c. Printing colors	Black, cyan, purple, magenta, red, yellow, green (colors can be specified down to the individual dot level)
d. Characters	Upper/lower case characters, numerals, symbols, and PET graphic characters
e. Character matrix	8 x 8 dot matrix
f. Character codes	CBM ASCII CODE
g. Character size	Height: 8 dots (0.984", 2.5mm) Width : 8 dots (0.866", 2.2mm)
h. Print speed	38 chars/sec
i. Max. number of columns	80 columns
j. Character spacing	10 characters/inch
k. Line spacing	1/6 and N/120 of an inch (software selectable)
l. Paper width	4.5 to 10" width (including tractor holes)
m. Multiple copies	Original + 1 copy (35 kg paper) (color is only on the original)
n. Graphics	Arbitrary combinations of any color of dot in an 8-dot column with up to 640 columns/line. Intermixing text and graphics on the same line is allowed.

2. Paper feed specifications

a. Paper feed direction	Forward direction only.
b. Minimum linefeed amount	1/120 of an inch
c. Linefeed speed	10 lines/sec (1/6" continuous linefeeds)

15 lines/sec(1/10" continuous linefeeds)

- d. Paper insertion From above at the back
- e. Paper feed method Both pin feed and friction feed are used together.

3. Ink ribbon specifications

- a. Ink ribbon Special cartridge type
- b. Ink color Black, cyan, magenta, yellow
4-color ribbon
- c. Ribbon life 1.2×10^6 characters

4. Operating Environment

- a. Power requirements 120V(USA,CANADA),220/240V(Europe)
AC+/-10%, 50/60Hz
- b. Power consumption 30 watts (continuous printing of text)
13 watts (standby)
- c. Temperature $5^{\circ}\text{C} - 35^{\circ}\text{C}$
- d. Humidity 20% - 80% (no condensation)
- e. External dimensions 477.5mm(W) x 348.5(D) x 141.8 (H)
(Without plastic rack)
- 6. Weight Approx. 5.2 kg

APPENDIX B: CBM ASCII CODE TABLE & FONT TABLE

A. Graphic Mode

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0 16 32 48 64 80 96 112 128 144 160 176 192 208 224 240	POS														
1	1 17 33 49 65 81 97 113 129 145 161 177 193 209 225 241	BUSINESS							GRAPHIC							
2	2 18 34 50 66 82 98 114 130 146 162 178 194 210 226 242	RVS ON							RVS OFF							
3	3 19 35 51 67 83 99 115 131 147 163 179 195 211 227 243	PAGE OFF							PAGE ON							
4	4 20 36 52 68 84 100 116 132 148 164 180 196 212 228 244	DC4														
5	5 21 37 53 69 85 101 117 133 149 165 181 197 213 229 245															
6	6 22 38 54 70 86 102 118 134 150 166 182 198 214 230 246															
7	7 23 39 55 71 87 103 119 135 151 167 183 199 215 231 247															
8	8 24 40 56 72 88 104 120 136 152 168 184 200 216 232 248															
9	9 25 41 57 73 89 105 121 137 153 169 185 201 217 233 249															
A	10 26 42 58 74 90 106 122 138 154 170 186 202 218 234 250	LF														
B	11 27 43 59 75 91 107 123 139 155 171 187 203 219 235 251	ESC														
C	12 28 44 60 76 92 108 124 140 156 172 188 204 220 236 252	FF FS														
D	13 29 45 61 77 93 109 125 141 157 173 189 205 221 237 253	CR							CS							
E	14 30 46 62 78 94 110 126 142 158 174 190 206 222 238 254	EN ON														
F	15 31 47 63 79 95 111 127 143 159 175 191 207 223 239 255	EN OFF														

Note) When an odd number of CHR\$(34) is detected in a line, the control codes \$00-\$1F and \$80-\$9F will be made visible by printing a reverse character for each of these controls. This will continue until an even number of quotes(CHR\$(34)) has been received or until end of this line.

B. Business Mode

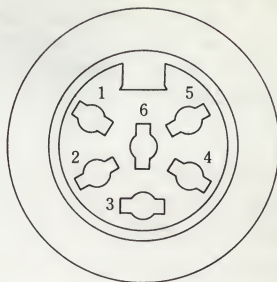
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0 16	32 48	64 80	96 112	128 144	160 176	192 208	224 240								
	POS															
1	1 17	33 49	65 81	97 113	129 145	161 177	193 209	225 241								
	BUSINESS								GRAPHIC							
2	2 18	34 50	66 82	98 114	130 146	162 178	194 210	226 242								
	RVS ON								RVS OFF							
3	3 19	35 51	67 83	99 115	131 147	163 179	195 211	227 243								
	PAGE OFF								PAGE ON							
4	4 20	36 52	68 84	100 116	132 148	164 180	196 212	228 244								
	DC4															
5	5 21	37 53	69 85	101 117	133 149	165 181	197 213	229 245								
6	6 22	38 54	70 86	102 118	134 150	166 182	198 214	230 246								
7	7 23	39 55	71 87	103 119	135 151	167 183	199 215	231 247								
8	8 24	40 56	72 88	104 120	136 152	168 184	200 216	232 248								
9	9 25	41 57	73 89	105 121	137 153	169 185	201 217	233 249								
A	10 26	42 58	74 90	106 122	138 154	170 186	202 218	234 250								
	LF															
B	11 27	43 59	75 91	107 123	139 155	171 187	203 219	235 251								
	ESC															
C	12 28	44 60	76 92	108 124	140 156	172 188	204 220	236 252								
	FF FS															
D	13 29	45 61	77 93	109 125	141 157	173 189	205 221	237 253								
	CR								CS							
E	14 30	46 62	78 94	110 126	142 158	174 190	206 222	238 254								
	EN ON															
F	15 31	47 63	79 95	111 127	143 159	175 191	207 223	239 255								
	EN OFF															

Note) When an odd number of CHR\$(34) is detected in a line, the control codes \$00-\$1F and \$80-\$9F will be made visible by printing a reverse character for each of these controls. This will continue until an even number of quotes(CHR\$(34)) has been received or until end of this line.

APPENDIX C: SERIAL INTERFACE INFORMATION

1. The Connector

Pin No.	Signal
1	SERIAL SRQ
2	GND
3	SERIAL ATN
4	SERIAL CLK
5	SERIAL DATA
6	<u>RESET</u>



2. The interface:

- Plug either end of the serial interface cable supplied with your printer into the Serial Bus connector for the connection. Refer to Chapter 1 Part H for details on how to connect your printer to your computer.
- When the printer is printing, no data will be transferred from the computer (the data line is said to be low). When the printer is at rest, and the cursor is flashing data can be transferred from the computer (the data line is said high).
- When a printer error occurs, all control circuits inside the Printer will stop.

APPENDIX D: HARDCOPY OF THE TEXT SCREEN

The following sample program can be used to get a hard printed copy of a program you have on your text screen. The Program is made to be used as a sub-routine. That means that when you use it, you must have a "GOSUB60000" in your program where appropriate.

```
60000 REM HARDCOPY OF TEXT SCREEN
60010 PO$=CHR$(16) :QT$=CHR$(34)
60020 RV$=CHR$(18) :RO$=CHR$(146)
60030 MF$=CHR$(145) :VR=PEEK(648)*256
60040 OPEN 4,4 :REM PRINT IN GRAPHIC MODE
60050 OPEN 6,4,6 :REM SET LINE SPACING
60060 PRINT#6,CHR$(12):REM CONTINUED VERTICAL DO
60070 FOR CL=0 TO 22 :QF=0:AS$=""
60080 FOR RO=0 TO 21
60090 SC=PEEK(VR+22*CL+RO)
60100 IF SC=34 THEN QF=1-QF
60110 IF SC>162 THEN 60140
60120 QF=1-QF:IF QF=1 THEN AS$=AS$+RV$+QT$:GOTO60200
60130 AS$=AS$+QT$+RO$:GOTO60200:GOTO60160
60140 IF QF=1 AND (SC>=128) THEN SC=SC-128:GOTO60160
60150 IF SC>=128 THEN SC=SC-128:RF=1:AS$=AS$+RV$
60160 IF SC<32 OR SC>95 THEN AS$=SC+64:GOTO60190
60170 IF SC>31 AND SC<64 THEN AS$=SC:GOTO60190
60180 IF SC>63 AND SC<96 THEN AS$=SC+32:GOTO60190
60190 AS$=AS$+CHR$(AS)
60200 IF RF=1 THEN AS$=AS$+RO$:RF=0
60210 NEXT RO
60220 IF QF=0 THEN PRINT#4,PO$"20"AS$:GOTO60240
60230 PRINT#4,PO$"20"AS$QT$
60240 NEXT CL
60250 PRINT#6,CHR$(20):REM ORIGINAL SPACING
60260 CLOSE4:CLOSE6
60270 RETURN
```

This program is made for Graphic Mode. If you wish to write the program in Business Mode, you must change OPEN 4,4 to OPEN 4,4,7 in line 60040.

This program is also made for VIC-20. If you wish to use this program to Commodore 64, you should change line 60070, 60080, 60090.

```
60070 FOR CL=0 TO 24 :QF=0:AS$=""
60080 FOR RO=0 TO 39
60090 SC=PEEK(VR+40*CL+RO)
```

Note) This program can't print the characters following the reverse quote.

APPENDIX E: MCS-801 GENERAL TEST PROGRAM

Following is the result (reduction copy: rate 70 %) of "MCS-801 General Test Program". Please enter the program part at first, and save this program to cassette or diskette. Then list up the program by the following command (please set first line to the papercutting line).

OPEN 4,4:CMD4,CHR\$(147):LIST

After listing of the program, close the printer file.

PRINT#4

CLOSE4

Then, type RUN and press RETURN key.

MCS-801 COLOR GRAPHIC PRINTER TEST

1. GRAPHIC MODE : SA=0

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0				0	@	P	-	7				r	-	7		r
1			!	1	A	Q	•	•				+	•	•		+
2			"	2	B	R		-			■	+		-	■	+
3			#	3	C	S	-	♥			-	+	-	♥	-	+
4			\$	4	D	T	-				-		-		-	
5			%	5	E	U	-	/					-	/		
6			&	6	F	V	-	X			■		-	X	■	
7			'	7	G	W		o				-		o		-
8			(8	H	X		•			■	-		•	■	-
9)	9	I	Y	~				7	-	~		7	-
A			*	:	J	Z	^	♦				7	^	♦		7
B			+	;	K	[^	+			+	-	^	+	+	-
C			,	<	L	£	L				■	■	L		■	■
D			-	=	M	J	\				L	7	\		L	7
E			.	>	N	†	/	π			7	■	/	π	7	■
F			/	?	O	+	Γ	▼			-	■	Γ	▼	-	■

2. business mode : sa=7

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0				0	@	p	-	P				r	-	P		r
1			!	1	a	q	A	Q				+	A	Q		+
2			"	2	b	r	B	R			■	+	B	R	■	+
3			#	3	c	s	C	S			-	+	C	S	-	+
4			\$	4	d	t	D	T			-		D	T	-	
5			%	5	e	u	E	U					E	U		
6			&	6	f	v	F	V			■		F	V	■	
7			'	7	g	w	G	W				-	G	W		-
8			(8	h	x	H	X			■	-	H	X	■	-
9)	9	i	y	I	Y			7	-	I	Y	7	-
a			*	:	j	z	J	Z				7	J	Z		7
b			+	;	k	[K	+			+	-	K	+	+	-
c			,	<	l	£	L				■	■	L		■	■
d			-	=	m	j	M				L	7	M		L	7
e			.	>	n	†	N	⊗			7	■	N	⊗	7	■
f			/	?	o	+	O	⊗			-	■	O	⊗	-	■

0
1
2
3
4

5
6
7
8
9

CHR\$(141)...CARRIAGE RETURN WITHOUT LINE FEED

```
ALL BIT PATTERN 1
ALL BIT PATTERN 1
ALL BIT PATTERN 2
ALL BIT PATTERN 2
```

```
0123456789012345678901234567890123456789012345678901234567890123456789  
01203456789001203456789001203456789001203456789001203456789001203456789  
01203456789001203456789001203456789001203456789001203456789001203456789  
01203456789 : ; < = ? @ 0123456789 ; ; < = ?  
@ A B C D E F G H I J K L M N O P Q R S T U V W X Y Z [ \ ] ^ _ ` ~ + , - . / : ; < = ?
```

[illegible]

10. CHR\$(147).....PAGING ON

11. CHR\$(17) & CHR\$(145) TEST

0 @ P - 7	r - 7	r	0 @ P - 7	r - 7	r	0 @ P - 7	r - 7	r
1 A Q - 8	t - 8	t	1 A Q - 8	t - 8	t	1 A Q - 8	t - 8	t
2 B R - 9	u - 9	u	2 B R - 9	u - 9	u	2 B R - 9	u - 9	u
3 C S - 0	v - 0	v	3 C S - 0	v - 0	v	3 C S - 0	v - 0	v
4 D T - 1	w - 1	w	4 D T - 1	w - 1	w	4 D T - 1	w - 1	w
5 E U - 2	x - 2	x	5 E U - 2	x - 2	x	5 E U - 2	x - 2	x
6 F V - 3	y - 3	y	6 F V - 3	y - 3	y	6 F V - 3	y - 3	y
7 G W - 4	z - 4	z	7 G W - 4	z - 4	z	7 G W - 4	z - 4	z
8 H X - 5	[- 5	[8 H X - 5	[- 5	[8 H X - 5	[- 5	[
9 I Y - 6] - 6]	9 I Y - 6] - 6]	9 I Y - 6] - 6]
: J Z - 7	_ - 7	_	: J Z - 7	_ - 7	_	: J Z - 7	_ - 7	_
; K [- 8	\ - 8	\	; K [- 8	\ - 8	\	; K [- 8	\ - 8	\
< L] - 9	^ - 9	^	< L] - 9	^ - 9	^	< L] - 9	^ - 9	^
= M _ - 0	` - 0	`	= M _ - 0	` - 0	`	= M _ - 0	` - 0	`
> N + - 1	~ - 1	~	> N + - 1	~ - 1	~	> N + - 1	~ - 1	~
? O - 2	- 2	- 2	? O - 2	- 2	- 2	? O - 2	- 2	- 2

0 @ P - 7	r - 7	r	0 @ P - 7	r - 7	r	0 @ P - 7	r - 7	r
1 A Q - 8	t - 8	t	1 A Q - 8	t - 8	t	1 A Q - 8	t - 8	t
2 B R - 9	u - 9	u	2 B R - 9	u - 9	u	2 B R - 9	u - 9	u
3 C S - 0	v - 0	v	3 C S - 0	v - 0	v	3 C S - 0	v - 0	v
4 D T - 1	w - 1	w	4 D T - 1	w - 1	w	4 D T - 1	w - 1	w
5 E U - 2	x - 2	x	5 E U - 2	x - 2	x	5 E U - 2	x - 2	x
6 F V - 3	y - 3	y	6 F V - 3	y - 3	y	6 F V - 3	y - 3	y
7 G W - 4	z - 4	z	7 G W - 4	z - 4	z	7 G W - 4	z - 4	z
8 H X - 5	[- 5	[8 H X - 5	[- 5	[8 H X - 5	[- 5	[
9 I Y - 6] - 6]	9 I Y - 6] - 6]	9 I Y - 6] - 6]
: J Z - 7	_ - 7	_	: J Z - 7	_ - 7	_	: J Z - 7	_ - 7	_
; K [- 8	\ - 8	\	; K [- 8	\ - 8	\	; K [- 8	\ - 8	\
< L] - 9	^ - 9	^	< L] - 9	^ - 9	^	< L] - 9	^ - 9	^
= M _ - 0	` - 0	`	= M _ - 0	` - 0	`	= M _ - 0	` - 0	`
> N + - 1	~ - 1	~	> N + - 1	~ - 1	~	> N + - 1	~ - 1	~
? O - 2	- 2	- 2	? O - 2	- 2	- 2	? O - 2	- 2	- 2

0 @ P - 7	r - 7	r	0 @ P - 7	r - 7	r	0 @ P - 7	r - 7	r
1 A Q - 8	t - 8	t	1 A Q - 8	t - 8	t	1 A Q - 8	t - 8	t
2 B R - 9	u - 9	u	2 B R - 9	u - 9	u	2 B R - 9	u - 9	u
3 C S - 0	v - 0	v	3 C S - 0	v - 0	v	3 C S - 0	v - 0	v
4 D T - 1	w - 1	w	4 D T - 1	w - 1	w	4 D T - 1	w - 1	w
5 E U - 2	x - 2	x	5 E U - 2	x - 2	x	5 E U - 2	x - 2	x
6 F V - 3	y - 3	y	6 F V - 3	y - 3	y	6 F V - 3	y - 3	y
7 G W - 4	z - 4	z	7 G W - 4	z - 4	z	7 G W - 4	z - 4	z
8 H X - 5	[- 5	[8 H X - 5	[- 5	[8 H X - 5	[- 5	[
9 I Y - 6] - 6]	9 I Y - 6] - 6]	9 I Y - 6] - 6]
: J Z - 7	_ - 7	_	: J Z - 7	_ - 7	_	: J Z - 7	_ - 7	_
; K [- 8	\ - 8	\	; K [- 8	\ - 8	\	; K [- 8	\ - 8	\
< L] - 9	^ - 9	^	< L] - 9	^ - 9	^	< L] - 9	^ - 9	^
= M _ - 0	` - 0	`	= M _ - 0	` - 0	`	= M _ - 0	` - 0	`
> N + - 1	~ - 1	~	> N + - 1	~ - 1	~	> N + - 1	~ - 1	~
? O - 2	- 2	- 2	? O - 2	- 2	- 2	? O - 2	- 2	- 2

12. CHR\$(18) & CHR\$(146) TEST

COMMODORE	COMMODORE	COMMODORE	COMMODORE	COMMODORE
COMMODORE	COMMODORE	COMMODORE	COMMODORE	COMMODORE
COMMODORE	COMMODORE	COMMODORE	COMMODORE	COMMODORE
COMMODORE	COMMODORE	COMMODORE	COMMODORE	COMMODORE
COMMODORE	COMMODORE	COMMODORE	COMMODORE	COMMODORE
COMMODORE	COMMODORE	COMMODORE	COMMODORE	COMMODORE

13. CHR\$(19).....PAGING OFF

Program list

```

1000 rem mcs-801 printer test
1010 en$=chr$(14) :de$=chr$(15)
1020 lf$=chr$(10) :cr$=chr$(13) :cs$=chr$(141)
1030 cd$=chr$(17) :cu$=chr$(145)
1040 rv$=chr$(18) :of$=chr$(146)
1050 po$=chr$(16) :esc$=chr$(27) :sub$=chr$(26)
1060 dc4$=chr$(20) :f$=chr$(28)
1070 cl$=chr$(19) :home$=chr$(147)
1080 :
1090 dim a$(15):for i=0 to 15:read a$(i):next
1100 data 0,1,2,3,4,5,6,7,8,9,a,b,c,d,e,f
1110 :
1120 open10,4,10:print#10:close10 :rem reset the printer
1130 :
1140 open4,4 :rem open printer file
1150 for i=1 to 41:print#4:next
1160 :
1170 :
1180 tl$="mcs-801 color graphic printer test "
1190 for i=1 to 3
1200 print#4,en$;
1210 for j=1 to len(tl$)
1220 co=co+1:if co=1 then co=2
1230 if co>7 then co=0
1240 print#4,dc4$chr$(co);mid$(tl$,j,1);
1250 next j
1260 print#4,de$;cs$;
1270 next i
1280 print#4:
1290 print#4,en$"_____ "de$
1300 print#4:print#4
1310 :
1320 print#4,dc4$;chr$(0);
1330 print#4,en$"1. graphic mode : sa=0 "de$
1340 gosub2420
1350 print#4:close4
1360 :
1370 open4,4,7
1380 print#4,dc4$;chr$(2);
1390 print#4,en$"2. business mode: sa=7 "de$
1400 gosub2420
1410 for i=1 to 3:print#4:next
1420 close4
1430 open4,4,0
1440 :
1450 print#4,dc4$;chr$(3);
1460 print#4,en$"3. chr$(10) test"de$:print#4
1470 for i=48 to 52:print#4,esc$po$chr$(1)chr$(24)chr$(i);lf$;:next
1480 for i=1 to 4:print#4:next
1490 :
1500 print#4,dc4$;chr$(4); :rem purple
1510 print#4:print#4,en$"4. chr$(13) test"de$:print#4
1520 for i=53 to 57:print#4,esc$po$chr$(1)chr$(24)chr$(i);cr$;:next
1530 :
1540 print#4,dc$;chr$(5); :rem green
1550 print#4:print#4,en$"5. chr$(141) test"de$:print#4
1560 for i=1 to 3
1570 print#4,"chr$(141)...carriage return without line feed"cs$;
1580 next
1590 print#4
1600 :
1610 print#4,dc4$;chr$(6); :rem blue
1620 print#4:print#4,en$"6. esc k n3n2n1 gd1gd2gd3...test"de$:print#4
1630 for j=1 to 2
1640 a$="" :for i=1 to 255:a$=a$+chr$(i):next
1650 print#4,esc$;"k";"255";a$;" all bit pattern 1"
1660 next
1670 for j=1 to 2
1680 b$="" :for i=255 to 1 step -1:b$=b$+chr$(i):next
1690 print#4,esc$;"k";"255"b$;" all bit pattern 2"
1700 next
1710 :

```



```

1720 print#4,dc4$;chr$(0); :rem black
1730 print#4:print#4,en$"7. chr$(14) & chr$(15) test"de$:print#4
1740 a$="0123456789"
1750 for i=1to8:print#4,a$;:next
1760 print#4
1770 for i=1to8:print#4,en$a$;:next
1780 print#4,de$
1790 for j=3to5
1800 a$="":for i=j#16toj#16+15:a$a$+chr$(i):next
1810 print#4,en$a$a$de$a$
1820 next
1830 :
1840 print#4,dc4$;chr$(2); :rem red
1850 print#4:print#4,en$"8. fs n gd test"db$:print#4
1860 hi=0
1870 for j=0to7:p=20+30*j#2;if p>256 then lo=p-255:goto1900
1880 print#4,fs$chr$(p)chr$(243):goto1910
1890 print#4,fs$chr$(lo)chr$(243):goto1910
1900 print#4,fs$chr$(255)chr$(243);fs$chr$(lo)chr$(243)
1910 next
1920 :
1930 print#4,dc4$;chr$(3); :rem blue
1940 print#4:print#4,en$"9. esc pos n1n2 test"de$:print#4
1950 open6,4,6:print#6,chr$(12):close6
1960 sp$=" "
1970 for i=0to360step10
1980 is=right$(sp$+str$(i),4)
1990 yo=220+120*asin(i%/180)
2000 yh=int(yo/256):yl=yo-yh#256
2010 print#4,i$esc$po$chr$(yh)chr$(yl)"#"
2020 next
2030 open6,4,6:print#6,chr$(20):close6
2040 for i=1to3:print#4:next
2050 :
2060 open3,4,3:print#3,chr$(52):close3
2070 print#4,dc4$;chr$(4); :rem purple
2080 print#4:print#4,en$"10. chr$(147).....paging on"de$
2090 print#4,home$; :rem paging on
2100 :
2110 print#4,dc4$;chr$(5); :rem green
2120 print#4,en$"11. chr$(17) & chr$(145) test"de$:print#4
2130 close4: open4,4,0
2140 gosub2600
2150 print#4
2160 close4
2170 open4,4,7
2180 gosub2600
2190 for i=1to3:print#4:next
2200 close4
2210 :
2220 open4,4
2230 print#4,dc4$;chr$(6); :rem blue
2240 print#4,en$"12. chr$(18) & chr$(146) test "de$:print#4
2250 a$=" commodore "
2260 for i=0to7:if i=1 then 2280
2270 print#4,dc4$;chr$(i);rvs$a$off$ "a$ "rvs$a$off$ "a$ "rvs$a$
2280 next
2290 for i=1to2:print#4:next
2300 :
2310 print#4,dc4$;chr$(2); :rem red
2320 print#4,en$"13. chr$(19).....paging off"de$
2330 print#4,cl$;
2340 :
2350 close4:open4,4,7
2360 open3,4,3:print#3,chr$(60):close3
2370 print#4,dc4$;chr$(0); :rem black
2380 for i=1to4:print#4:next
2390 cmd4,home$;en$"program list"de$:list
2400 end
2410 :
2420 v$=chr$(221):s$=chr$(32):tv$=s$+v$+s$:
2430 open6,4,6:print#6,chr$(12):close6
2440 print#4,"

```

```

2450 print#4,u$$$$$;:for i=0to15
2460 print#4,s$u$$$a$(i);
2470 next:print#4,s$u$
2480 for i=0to15
2490 print#4," |-----|
| "
2500 print#4,u$$$a$(i)tu$$$tu$$$tu$chr$(i+32)tu$chr$(i+48)tu$chr$(i+64);
2510 print#4,tu$chr$(i+80)tu$chr$(i+96)tu$chr$(i+112)tu$$$tu$$$tu$;
2520 print#4,chr$(i+160)tu$chr$(i+176)tu$chr$(i+192)tu$chr$(i+208);
2530 print#4,tu$chr$(i+224)tu$chr$(i+240)s$u$
2540 next
2550 print#4," |-----|
| "
2560 print#4
2570 open6,4,6:print#6,chr$(20):close6
2580 return
2590 :
2600 sp$=" ":for i=0to15
2610 a$=""
2620 a$=a$+chr$(i+48)+sp$
2630 a$=a$+chr$(i+64)+sp$+chr$(i+80)+sp$
2640 a$=a$+chr$(i+96)+sp$+chr$(i+112)+sp$
2650 a$=a$+chr$(i+160)+sp$+chr$(i+176)+sp$
2660 a$=a$+chr$(i+192)+sp$+chr$(i+208)+sp$
2670 a$=a$+chr$(i+224)+sp$+chr$(i+240)
2680 print#4,a$" "cd$a$" "cu$a$
2690 next
2700 return

ready,

```

MEMO

MEMO

MEMO

COMMODORE SALES CENTERS

Commodore Business Machines, Inc.

1200 Wilson Drive
Westchester, PA 19380, U.S.A.

Commodore Business Machines Ltd.

3370 Pharmacy Avenue, Agincourt,
Ontario, M1W 2K4, Canada

Commodore Business Machines (UK) Ltd.

675 Ajax Avenue, Trading Estate,
Slough Berks, SL1 4BG, England

Commodore Bueromashinen GmbH

Lyonerstrasse 38, PO BOX 710126
Frankfurt 6000, West Germany

Commodore Italiana S.R.L.

Via Conservatoria 22
Milano 20122, Italy

Commodore Information Center

5 Onion Road
Lane Cove 2066 NSW, Australia

Commodore Computer B.V.

Marksingel 2e4811 N.V. Breda
Postbus 720, 480 3aS Breda, Netherlands

Commodore AG(Schweiz)

Aeschenvorstadt 57
4010 Basel, Switzerland

DISK DRIVES

MODEL 1541

Media compatible with 4040 and 2031LP diskettes.
170 K of memory per single side disk.
Serial port interface.

MODEL SFD 1001

1 Megabyte double sided, floppy disk drive.
Uses double density, double sided diskettes.
IEEE interface.

MODEL 2031LP

Media compatible with 1541 and 4040 diskettes.
170 K of memory per single side disk.
Rugged and stylish housing.
IEEE interface.

PRINTERS

MODEL MPS-801

High quality dot matrix printing.
80 column printing.
Prints 50 characters per second.
Serial port interface.

MODEL MPS-802

High quality dot matrix printing.
80 column printing.
Prints 80 characters per second.
Perfect for everything from personal letters to important business reports.
Serial port interface.

PRINTER/PLOTTER

MODEL 1520

Four color, printer/plotter for all graphic needs.
Print bar graphs...create pie charts...plot time graphs.
High quality printing in a variety of styles and sizes.
Great for graphic artists.

MODEMS

VICMODEM

Timely financial information, news and reference libraries are as near as your phone.
An affordable introduction to the growing world of phone computer services.
Communicate with a wide variety of computer users who already own a modem.

AUTOMODEM

The convenience of automatic answering and dialing.
The resource power of telecomputing services.
Turn your phone into a telecomputing information network.

